RESIDENT SPACE OBJECT LIGHT CURVE CLASSIFICATION & SPACE SITUATIONAL AWARENESS SENSITIVITY & SIMULATION STUDIES

RANDA QASHOA

A DISSERTATION SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN EARTH AND SPACE SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

NOVEMBER 2024

Abstract

The number of objects being launched into space is rapidly increasing, emphasizing the critical importance of detecting, characterizing, and tracking these objects—an area of focus known as Space Situational Awareness (SSA). These Resident Space Objects (RSOs) include satellites (both active and inactive), rocket bodies and debris. Knowing the type of object near our satellites of interest is very important as it gives satellite operators the knowledge needed to accurately plan maneuvers to keep our orbits safe. This dissertation explores three main contributions within the field of SSA. The first is a light curve classifier which uses Machine Learning (ML) to classify Low Earth Orbit (LEO) RSOs into stable satellites, tumbling satellites and rocket bodies. Multiple approaches were tested but the method with the highest accuracy is a Barlow Twins network which has a 75% accuracy for two minute light curves and a 97% accuracy for five minute light curves. The classification is used to characterize the motion of objects, which operators can use in combination with real images to determine the risk of collision and to perform effective maneuvers. The second contribution is regarding SSA mission planning. A sensitivity analysis was conducted to determine the best camera to use for observing co-orbiting RSOs within 250 km of the observer. The analysis includes exploring the location of potential targets in the Field-Of-View (FOV) of the observer as well as the Signal-to-Noise Ratio (SNR) of different targets. A similar analysis to the one presented in this dissertation has been performed for the Redwing microsatellite mission. Lastly, RSO image prediction simulations are tested for use in SSA. This dissertation demonstrated the implementation of an anti-sun pointing orientation for prediction simulations with validation from real images. Predicted images were used to determine targets for observation which were then validated following the downlink of the images.

Dedication

“That which we need the most will be found where we least want to look.” —Carl Jung.

To my friends in the Nanosatellite Research Lab, thank you for you constant feedback and support. I would not have been able to do this without you.

"It’s kind of fun to do the impossible." —Walt Disney

To my sister, Nada, thank you for listening to my thesis rants. You are truly a poor unfortunate soul for bearing through that.

"What progress, you ask, have I made? I have begun to be a friend to myself." —Seneca, philosopher

To everyone who thinks they are not good enough, if I could do it then so can you.

Acknowledgements

There are many people in my life that I am grateful for. Working on this Ph.D. was not an easy journey, but it was much more tolerable due to the assistance and support I received from my family, friends, and colleagues.

I would first like to thank my supervisor, Dr. Regina S.K. Lee. You are truly like a mother to me. I was able to learn and grow so much as a result of the opportunities you have granted me, and I will forever be thankful for that. Going to a diner or a breakfast restaurant reminds me of the many experiences we shared at places like that. You can never go wrong with a triple chocolate pancake.

I would also like to thank Dr. Ryan Clark, my mentor and inspiration for pursuing a Doctoral degree. I have learned so much from you, Ryan, regarding technical and interpersonal matters. Thank you for all the support you've given me during the pursuit of my graduate degree. You are the best mentor I could ever ask for.

I would also like to thank my closest friends at the Nanosatellite Research Lab. We have been through so much and it only made me stronger. I am immensely thankful for everyone so I will list this in no particular order. I would like to thank Gabriel Chianelli and Perushan Kunalakantha (RPG). I have learned so much about technical matters and the world because of you. All the discussions we've had about reference frames (with Gabe) and Barlow Twins (with Perushan) were immensely helpful for putting together this dissertation. Also, all the hours we spent planning the RSONAR II mission will never be forgotten.

I would like to thank my fellow female labmate, Andrea Vallecillo Baires (PRAGmatic). I always enjoy our talks. You are my first non-family travel buddy, and I had a blast on all our work trips. Next, I would like to thank all the RSONAR II Flight Captains (RPG + Vithurshan Suthakar). It was not an easy mission and there were significant roadblocks along the way, but the instant STARDUST worked is one of the best moment in my life. While things have changed since that time, Timmins Randy will forever live in my heart. I would also like to thank Michael Fairbrother (GRaMP). Seeing all the cool coding projects you've done and your thesis work on deep reinforcement learning is really inspiring.

I have had so many "firsts" in the lab and it's all thanks to Dr. Regina Lee and my friends. You are all the best! If anyone is wondering where all these random acronyms came from, you can thank Perushan for those. He's our lab's acronym master.

Last but certainly not least, I would like to thank my sister, Nada for all her help. Thank you for listening to all my video game and thesis rants even if you didn't follow up with most of it. Talking about it helped. Thank you for letting me drag you around to different locations across the world when all you wanted to do was chill in a hotel room all day. I had a lot of fun exploring new cities and trying out new experiences. While you are my only sister, you are also my favorite sister. Thanks for everything sis!

Table of Contents

List of Tables

List of Figures

Preface

- The work in chapter 2 is based on the following publication: R. Qashoa and R. S. K. Lee, "Classification of Low Earth Orbit (LEO) Resident Space Objects' (RSO) Light Curves Using a Support Vector Machine (SVM) and Long Short-Term Memory (LSTM)," *Sensors 2023, Vol. 23, Page 6539*, vol. 23, no. 14, p. 6539, Jul. 2023, ISSN: 1424-8220. DOI: 10.3390/S23146539. [Online]. Available: https://www.mdpi.com/1424-8220/23/14/6539.

- Chapter 3 is currently under review as R. Qashoa and R. S. K. Lee, "Low Earth Orbit (LEO) Resident Space Object (RSO) Light Curve Classification using Barlow Twins," *IEEE Transactions on Aerospace and Electronic Systems*, Under Review 2024.

- Chapter 5 is based on the following publication: R. Qashoa *et al.*, "Wide Field of View (FOV) Imagers for Co-Orbiting Object Detection," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2024. [Online]. Available: https://amostech.com/TechnicalPapers/2024/Space-Based-Assets/Qashoa.pdf.

- Chapter 6 is based on the following publication: R. Qashoa *et al.*, "RSO Simulations with Anti-Sun Pointing Predictions," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2023. [Online]. Available: https://amostech.com/TechnicalPapers/2023/Poster/Qashoa.pdf.

- Chapter 7 is currently under review as R. Qashoa *et al.*, "Demonstration of RSO Simulation, Prediction and Detection Using a Low-Resolution Imager in Anti-Sun Pointing," *Advances in Space Research*, Under Review 2024.

As noted above, the dissertation contains content that has been published in several conferences and peer-reviewed articles. In addition to the aforementioned references, the following lists provides presentations and publications relevant to the research presented here, but not directly provided as chapters.

- A. Porras-Hermoso *et al.*, "Accuracy analysis of Space Based Space Debris Tracking for objects in a Low Earth Orbit," *Advances in Space Research*, Under Review 2024.
- R. Qashoa *et al.*, "Lessons Learned From Space Situational Awareness Payloads On Stratospheric Balloon," in *Canadian Aeronautics and Space Institute (CASI) ASTRO*, 2024.
- R. Qashoa *et al.*, "Lessons Learned From Technology Demonstration Of Space Situational Awareness Satellite," in *Small Satellite Conference*, 2024.
- R. Qashoa and R. S. K. Lee, "Comparison of Low Earth Orbit (LEO) Resident Space Object (RSO) Artificial Intelligence (AI) Light Curve Classification Techniques," in *Committee on Space Research (COSPAR) 45th Scientific Assembly*, 2024.
- P. Kunalakantha *et al.*, "Resident Space Object (RSO) Tracking in Space-Based, Low Resolution, Non-Constant-Attitude Imagery," *Remote Sensing*, Under Review 2024.
- R. Qashoa *et al.*, "Technology Demonstration of SSA Mission on Stratospheric Balloon Platform," *Remote Sensing*, vol. 16, no. 5, p. 749, Feb. 2024. DOI: `10.3390/rs16050749`. [Online]. Available: `https://www.mdpi.com/2072-4292/16/5/749`.
- R. Qashoa *et al.*, "SPACEDUST-Optical: Wide-FOV Space Situational Awareness from Orbit," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2023. [Online]. Available: `https://amostech.com/TechnicalPapers/2023/Poster/Qashoa2.pdf`.
- V. Suthakar *et al.*, "Comparative Analysis of Resident Space Object (RSO) Detection Methods," *Sensors*, vol. 23, no. 24, 2023. DOI: `10.3390/s23249668`.
- P. Harrison *et al.*, "Utilizing the CASSIOPE FAI Instrument for Resident Space Object Monitoring," in *Division of Atmospheric and Space Physics (DASP) Workshop*, 2023.
- R. Qashoa *et al.*, "Assessing Resident Space Objects (RSO) Stability Using Light Curves," in *Canadian Aeronautics and Space Institute (CASI) ASTRO*, 2022.
- R. Qashoa, "Light Curve Analysis Of Resident Space Objects," in *NSERC-CREATE TEPS Conference*, 2022.

All other parts of this thesis are the original work of Randa Qashoa, and have not been previously published.

List of Acronyms

Acronym	Description
1D	1-Dimensional
ADAM	Adaptive Moment
AI	Artificial Intelligence
CAD	Computer-Aided Design
CASSIOPE	Cascade, Smallsat and Ionospheric Polar Explorer
CCA	Connected Components Analysis
CMOS	Complementary Metal Oxide Semiconductor
CNES	Centre National D'Etudes Spatiales
CNN	Convolutional Neural Network
CoM	Center of Mass
COTS	Commercial Off-The-Shelf
CSA	Canadian Space Agency
Dec	Declination
DPSN	Dual Prototypical Shapelet Networks
DRDC	Defence Research & Development Canada
DWT	Discrete Wavelet Transformation
ECI	Earth Centered Inertial
EMC	Electro-Magnetic Compatibility
EOIR	Electro-Optical Infrared
e-POP	enhanced Polar Outflow Probe
ESA	European Space Agency

Acronym	Description
FAI	Fast Auroral Imager
FEM	Finite Element Modeling
FFT	Fast Fourier Transformation
FN	False Negatives
FOV	Field-Of-View
FP	False Positives
FPGA	Field Programmable Gate Array
FWHM	Full Width at Half Maximum
GEO	Geostationary Orbit
Grad-CAM	Gradient-weighted Class Activation Mapping
HMM	Hidden Markov Model
IOD	Initial Orbit Determination
LAMB	Layer-wise Adaptive Moments optimizer for Batch
LEO	Low Earth Orbit
LIME	Local Interpretable Model-agnostic Explanations
LIS	Lost In Space
LLA	Latitude, Longitude, and Altitude
LSTM	Long Short-Term Memory
MEO	Medium Earth Orbit
ML	Machine Learning
NEOSSat	Near Earth Object Surveillance Satellite
OBC	On-Board Computer
OMM	Orbital Mean-Elements Message
PCA	Principal Component Analysis
PCB	Printed Circuit Board
PDM	Phase Dispersion Minimization
PDU	Power Distribution Unit
ProSeNet	Prototype Sequence Network
QUEST	Quaternion Estimator

Acronym	Description
ReLu	Rectified Linear Unit
ResNet	Residual Network
RF	Random Forest
RA	Right Ascension
RNN	Recurrent Neural Network
RSO	Resident Space Object
RSONAR	Resident Space Object Near-Space Astrometric Research
SBOIS	Space-Based Optical Image Simulator
SBSS	Space-Based Space Surveillance
SB-SSA	Space-Based Space Situational Awareness
sCMOS	scientific Complementary Metal Oxide Semiconductor
SDA	Space Domain Awareness
SHAP	SHapley Additive exPlanations
SNR	Signal-to-Noise Ratio
SoC	System on Chip
SPACEDUST	Special Processes and Advanced Computing Environment for Detection, Unambiguity, Surveying, and Tracking
SSA	Space Situational Awareness
SSH	Secure Shell
SSN	Space Surveillance Network
STARDUST	Star Tracker Attitude and RSO Detection for Unified Space Technologies
STK	Systems Tool Kit
STM	Space Traffic Management
SVM	Support Vector Machine
TIMED	Thermosphere Ionosphere Mesosphere Energetics and Dynamics
TLE	Two-Line Element
TP	True Positives
t-SNE	t-distributed Stochastic Neighbor Embedding
UHS	Ultra-High Speed

Acronym	Description
USB	Universal Serial Bus
UTC	Universal Time Coordinated
VAE	Variational Auto Encoder
WCS	World Coordinate System
WST	Wavelet Scattering Transformation
XAI	eXplainable Artificial Intelligence

List of Symbols

Symbol	Description
θ	Angular difference
θ_{pixel}	Instantaneous FOV
ρ	Observed target's reflectivity
τ	Optical transmittance
τ_{atm}	Atmospheric transmittance
$\phi(t)$	Low pass filter
Φ	Solar phase angle
$\psi(t)$	Mother wavelet
ω	Angular rate
A	Aperture area
A_{target}	Area of observed target
e_b	Number of background photoelectrons
e_{DC}	Dark current noise
e_n	Read noise
e_s	Number of signal photoelectrons
d_{target}	Observed target's diameter
$D_{aperture}$	Aperture diameter
E_{RSO}	Photon-based irradiance
$Error_{CrossTrack}$	Cross track error
$F(\Phi)$	Function relating illumination phase angle to fraction of reflected light
L_b	Background signature

Symbol	Description
m_{obj}	Visual magnitude of observed target
m_{sun}	Visual magnitude of sun (value = -26.74)
M_b	Background visual magnitude per square arcsec
n	Total number of data points (or light curves)
P_{ij}	Pairwise similarity between two points in the original data space
Q_{ij}	Pairwise similarity between two points in the lower dimension 3D data space
Q_e	Quantum Efficiency
$\mathbf{r_{bdy}}$	Position of RSO in body frame
$\mathbf{r_{host-ECI}}$	Position of the host spacecraft in ECI frame
$\mathbf{r_{orb}}$	Position of RSO in orbital frame
$\mathbf{r_{sun-ECI}}$	Position of the sun in ECI frame
$\mathbf{r_{target-ECI}}$	Position of observed target in ECI frame
R	Range (distance between observer and target)
$\mathbf{R}$	Rotation matrix
$s(t)$	Input signal
t	Camera integration time
t_{signal}	Signal integration time
w	Weight vector
W_s	Wavelet scattering coefficients
x_i	Network input
X	Position along X coordinate
y_i	Classification output class
Y	Position along Y coordinate
Z	Position along Z coordinate
$\mathbf{V}$	Position vector

Chapter 1

Introduction

The trend of launching more satellites into space, particularly nanosatellite constellations, is currently on the rise. As a result, there has been a dramatic growth in the number of Resident Space Objects (RSOs), which include satellites, rocket bodies, and debris. This increase in the number of RSOs is visualized in Figure 1.1 from the European Space Agency (ESA) [17].

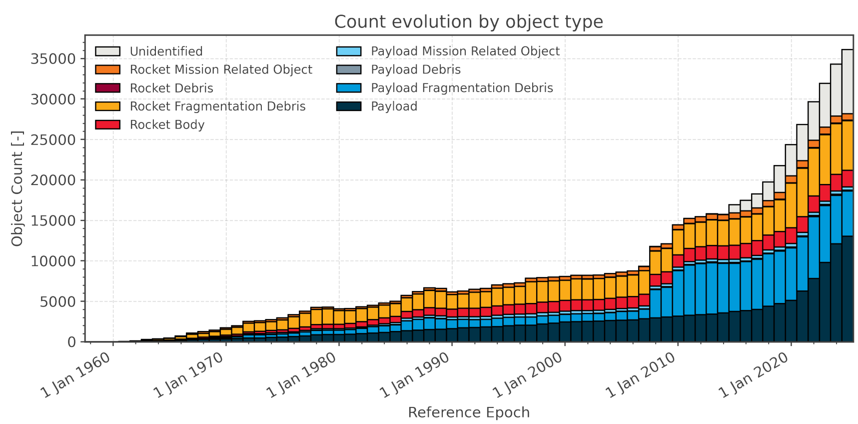


Figure 1.1: Number of RSOs around Earth from 1960 to 2023. Source: [17]

As the number of satellites launched into space continues to rise, ensuring their safety from collisions with other RSOs is becoming increasingly critical and challenging. Past collisions between RSOs highlight the severity of this issue. The most notable of which is the 2009 collision between the operational Iridium 33 and the non-functional Cosmos 2251 satellites which created more than 1800 new debris tracked by the Space Surveillance Network (SSN) [18]. More recently, in 2024, the Thermosphere Ionosphere Mesosphere

Energetics and Dynamics (TIMED) and Cosmos 2211 satellites almost collided with less than 10 m between the spacecraft [19]. These incidents demonstrate the critical role of Space Situational Awareness (SSA), which involves the detection, tracking, and characterization of RSOs. SSA is crucial for identifying RSOs in dangerous proximity to satellites and estimating their motion to execute necessary maneuvers, thereby avoiding potential collisions.

SSA is a field of global interest with a 2030 global market size forecast of $2.68 Billion USD increasing from $1.88 Billion USD in 2023 [20]. On a global scale, North America generated approximately 40% of the global market share, and is expected to continue to maintain its dominance in the SSA industry [20]. In Canada, significant contributions are also being made to SSA research. Defence Research & Development Canada (DRDC), on behalf of the government, is actively conducting numerous research studies in this field. Canada has two SSA satellites: Sapphire and Near Earth Object Surveillance Satellite (NEOSSat) with a third one, named Redwing, currently in development. The analysis in Chapter 5 was performed as a research activity for the Redwing project. DRDC is analyzing data from NEOSSat as a space-based observer while also using a ground-based observatory to support NEOSSat's operations, as well as other SSA activities [21].

To capture the needed RSO data for SSA research, both ground-based and space-based imagers are used. Ground-based imaging is more accessible due to the ease of setting up and operating telescopes, though this method comes with limitations. The data collected by ground-based observatories are limited by the RSOs and stars that are in the Field-of-View (FOV) due to the geographic location of the camera. Additionally, the images captured from ground-based cameras are impacted by atmospheric attenuation [22]-[23]. Space-based imaging offers more diverse RSO capture capabilities due to the camera being mounted on a moving platform. However, spacecraft development takes longer and is more expensive than building ground stations. Additionally, space-based imagers face risks such as potential collisions, and unlike ground-based cameras, they cannot be modified once launched. On top of this, the data obtained is constrained by downlink rates. Despite these challenges, the improved imaging quality and flexibility of space-based imagers justify their use. As a result, the term Space-Based Space Situational Awareness (SB-SSA) has gained increased usage due to its significance.

This dissertation demonstrates a ground-based photometric light curve classification algorithm as a main contribution. Additional contributions include wide FOV camera analysis and space-based image simulations for SSA missions.

With regard to photometric analysis, light curves are time series data of brightness. In this research, light curves refer specifically to the brightness time series of RSOs, excluding astronomical light curves of stars or asteroids. Light curves provide valuable information about RSOs, including their shape, material composition, type, and spin rate. As a result, this 1-Dimensional (1D) data are especially beneficial for collision avoidance. Knowledge of an RSO's spin rate provides insight into its motion, which allows operators to prioritize targets with irregular or unpredictable movements for collision avoidance maneuvers. Furthermore, knowledge of an RSO's shape and surface properties allow us to characterize the RSO. Light curves require much fewer resources than images, so the use of light curves allows for more efficient data storage where only the necessary information is used for the analysis. One of the first steps in light curve analysis is understanding the type of object being observed in order to understand its motion. RSOs can be satellites, rocket bodies or debris. Even when looking at satellites, their motion varies depending on whether it is an actively controlled spacecraft or an inactive, and therefore, tumbling object. Analyzing a large number of light curves can be challenging, as each curve requires individual tuning. A common approach for batch light curve classification is to apply Machine Learning (ML) techniques to categorize RSOs based on their light curves. This is important for ground-based satellite operators, as it helps them understand objects orbiting near their assets. Furthermore, with the development of an efficient light curve classifier, light curve extraction and classification could potentially be performed in orbit as a first step in motion characterization of nearby objects.

Developing missions for SB-SSA is a rapidly expanding field. An important aspect of SSA mission planning is selecting the appropriate detector to use. With numerous vendors available, the optimal choice depends on the mission objectives. There is a need for analysis of the new RSO's orbit, as well as the different camera options to assess the best camera to accomplish objectives. Wide FOV cameras are typically used for scanning large swaths of the sky, which is extremely beneficial for astrometric SSA, whereas photometric analysis typically relies on narrow FOV cameras with better imaging capability. This dissertation will focus on camera selection for an SSA mission primarily designed for astrometric analysis, with additional capabilities for photometric analysis.

Astrometric analysis uses images to determine an RSO's orbit and motion. With RSO images, we can predict, detect, and track RSOs. For this dissertation, the RSOs are treated as point sources in the images as the focus is on the position of RSOs in orbit. A significant challenge of SSA, particularly SB-SSA, is the restricted access to RSO images. These restrictions come from national security concerns, commercial technology protection, and geopolitical sensitivities, making it difficult for researchers to obtain the needed images. Consequently, image simulations have become increasingly important for mission planning, RSO orbit prediction, validating SSA algorithms for star and RSO detection, and generating light curves. Simulated images enable us to estimate how an image from a specified observer looks like with accurate star and RSO pixel locations that are used for the testing and development of other SSA algorithms. Additionally, for photometric analysis missions, image simulations facilitate the creation of light curves, the testing of light curve analysis algorithms, and the determination of optimal camera pointing to capture the highest quality light curves. For SB-SSA, generating simulated images that accurately represent the RSOs visible to an observer requires precise knowledge of the observer spacecraft's attitude. Unfortunately, in many scenarios, detailed attitude information is unavailable. To remedy this, we can model common satellite pointing profiles and simulate those. Examples of common pointing profiles for optical cameras include target-centering, anti-sun pointing, nadir, and inertial pointing. With regard to light curve extraction, anti-sun pointing provides favorable conditions where the targets appear bright and the camera is not saturated by the sun. An example of an imager in anti-sun pointing is the Fast Auroral Imager (FAI) onboard the Cascade, Smallsat and Ionospheric Polar Explorer (CASSIOPE). CASSIOPE has publicly available data that can be used for validating image simulation algorithms.

1.1 Motivation

This dissertation addresses gaps in both photometric and astrometric techniques in SSA. While light curves are crucial for photometric analysis, they present challenges due to their complexity. In comparison to extensively researched astronomical light curves, RSO light curve studies are relatively new. Most RSO light curve analysis studies are performed on Geostationary (GEO) RSOs, yet the majority of RSOs are in Low Earth Orbit (LEO). LEO RSO light curves are challenging to study as they are short in duration imaging opportunities. LEO RSOs play a critical role in our daily lives for communication, navigation, and weather observations. Therefore, understanding their motion and behavior in space is essential for ensuring their safety in orbit. Analyzing light curves helps to determine the motion and stability of RSOs, which is highly beneficial for the SSA community.

One aspect to consider in SSA mission design is choosing the hardware to best achieve mission objectives. In general, when selecting a camera for SSA, key factors include star and RSO observability. When building missions for photometric analysis, the brightness of observed RSOs also plays a major role. However, most camera data sheets do not provide direct values for assessing star and RSO observability or brightness estimates. To address this, a sensitivity analysis algorithm was developed to evaluate the performance of different cameras in observing nearby RSOs.

Additionally, numerous algorithms exist in the literature that independently predict [24, 25], simulate [26, 27], detect [13, 28], and identify RSOs [29, 30]. A secondary focus of this thesis is to demonstrate a pipeline applied to a LEO RSO, as part of a feasibility study with collaborators. This pipeline begins with RSO prediction and concludes with its detection. The contribution of this work is an improved RSO simulator that has a new pointing mode to simulate anti-sun pointing conditions. This assists with photometric analysis. This new pointing mode allows us to simulate a wide variety of RSOs and enable RSO prediction for scenarios where measured attitude data is not available. An end-to-end SSA system is beneficial for future SSA missions, and this feasibility study highlights the viability of such a pipeline for future missions.

More details on light curves, sensitivity analysis, and image simulations are provided in Chapters 2 - 7.

1.2 Research Outline

The three research projects in this dissertation are applied at different phases of a space mission and address the following issues:

- Pre-mission Launch – Detectability Analysis:
  - An important aspect in SSA mission planning is deciding the correct optical camera to accomplish mission objectives.
  - There is a need for analysis of the new RSO's orbit as well as the different camera options to determine the best camera.
  - Most camera data sheets do not offer direct values that can be used to assess star and RSO observability as well as brightness estimates.
- Pre/During mission operation – Image simulation and predictions:
  - Image simulations have become increasingly important for mission planning and validating SSA algorithms.
  - Knowledge of the observer spacecraft's attitude is key to generating simulations, but this data is rarely available.
- Post data collection – LEO Light curve classification:
  - Light curves (measurements of RSO reflectivity over time) allow for RSO characterization. 84% of satellites are in LEO but most studies are performed for GEO satellites.
  - This is due to LEO light curves being shorter than GEO, so they are more challenging to study.

1.3 Research Objectives

The main objectives of this research are to demonstrate a photometric analysis RSO classifier for characterizing space objects and to simulate RSOs for missions. This dissertation presents the tools for light curve classification (used in photometric studies), optical system sensitivity analysis, and RSO prediction with anti-sun pointing via simulation. The detailed research objectives are as follows:

- Develop and validate an algorithm for LEO RSO light curve classification
- Develop a tool for performing sensitivity analysis to assess the performance of various cameras for co-orbiting object detection in SSA missions
- Implement RSO prediction using simulations in anti-sun pointing mode

1.4 Dissertation Outline

PART I: Light Curve Classification, which is the main contribution in this thesis, describes various ML algorithms that were developed for LEO RSO light curve classification. The results from these different algorithms are compared. Chapter 2 shows three approaches used for a supervised learning light curve analysis study. Chapter 3 shows an improved ML algorithm that addresses the shortcomings of the algorithms in Chapter 2 by using a self-supervised approach. Chapter 4 offers a qualitative study to understand the performance of the light curve classifier. PART II: SSA Payload & Simulation Study illustrates the utilization of a low spatial resolution and wide FOV imager for SSA applications. Chapter 5 provides an explanation of a sensitivity analysis algorithm, used in a mission analysis study, to choose an appropriate camera for detection of co-orbiting objects at certain threshold distances away from our spacecraft of interest. This analysis was applied during the Redwing microsatellite mission development, but can also be used for future SSA missions. Chapter 6 explains an anti-sun pointing mode that was added to the Space-Based Optical Image Simulator (SBOIS) [31], to allow for RSO prediction when detailed attitude knowledge is not available. Anti-sun pointing allows for favorable imaging conditions for photometric analysis, and the performance can be validated with FAI, which is currently locked in anti-sun pointing. Chapter 7 shows a series of steps to go from RSO prediction to image simulation using the new pointing mode, and then to real-image RSO detection. Such a pipeline has been tested with the FAI as well, but can be applied in future SSA missions. Chapter 8 provides overall conclusions from the studies in PARTS I-II and provides a list of future work that this dissertation will lead to.

PART I: Light Curve Classification

This part has three chapters demonstrating techniques for LEO light curve classification. In Chapter 2, three approaches to supervised learning classification are described. This chapter is based on the paper R. Qashoa and R. S. K. Lee, "Classification of Low Earth Orbit (LEO) Resident Space Objects' (RSO) Light Curves Using a Support Vector Machine (SVM) and Long Short-Term Memory (LSTM)," *Sensors 2023, Vol. 23, Page 6539*, vol. 23, no. 14, p. 6539, Jul. 2023, ISSN: 1424-8220. DOI: `10.3390/S23146539`. [Online]. Available: `https://www.mdpi.com/1424-8220/23/14/6539`. There are two limitations with the approaches in Chapter 2: the first is that the networks take in features from the light curves as opposed to the light curves themselves. The second limitation is that the light curves need to be five minutes in duration which means many shorter light curves had to be excluded from classification.

To address both concerns, an improved light curve algorithm was developed. This is presented in Chapter 3 and is currently under review as R. Qashoa and R. S. K. Lee, "Low Earth Orbit (LEO) Resident Space Object (RSO) Light Curve Classification using Barlow Twins," *IEEE Transactions on Aerospace and Electronic Systems*, Under Review 2024. The results in Chapter 3 include a comparison with the best-performing algorithm in Chapter 2. Chapter 3 offers a comparison of algorithm performance for two, three, and five minute long light curves to assess how well the different classifiers perform when we are limited by very short light curves. The dataset in Chapter 3 is the same as the one in Chapter 2 and the information regarding the new method is presented in Section 3.4.

To gain qualitative insight into the performance of the Barlow Twins network, Chapter 4 demonstrates the use of eXplainable AI (XAI) for analyzing the outputs of light curve classification. The results from this chapter have not been published elsewhere. As a result, this chapter does not include its own abstract.

Chapter 2

Classification of Low Earth Orbit Resident Space Object Light Curves Using a Support Vector Machine and Long Short-Term Memory

Abstract

Light curves are plots of brightness measured over time. In the field of SSA, light curves of RSOs can be utilized to infer information about an RSO such as the type of object, its attitude, and its shape. Light curves of RSOs in GEO have been a main research focus for many years due to the availability of long time series data spanning hours. Given that a large portion of RSOs are in LEO, it is of great importance to study trends in LEO light curves as well. The challenge with LEO light curves is that they tend to be short, typically no longer than a few minutes, which makes them difficult to analyze with typical time series techniques. This study presents a novel approach to observational LEO light curve classification. We extract features from light curves using Wavelet Scattering Transformation (WST) which is used as an input for an ML classifier. We performed light curve classification using both a conventional ML approach, namely a Support Vector Machine (SVM), and a deep learning

technique, Long Short-Term Memory (LSTM), to compare the results. LSTM outperforms SVM for LEO light curve classification with a 92% accuracy. This proves the viability of RSO classification by object type and spin rate from real LEO light curves.

2.1 Introduction

This section provides background information on light curves, the dataset we are using, and the methods we are applying to obtain features from light curves in order to classify them.

2.1.1 RSO Light Curves

Light curves represent the brightness of objects (stars, RSOs and planets) as a function of time. They are commonly used in various space applications (astronomy, planetary studies, etc.) to understand the characteristics of celestial objects. Various studies on SSA, such as [32–36], rely on light curve information to extract the properties of RSOs in the trends of this time series data; as such, extracting RSO features from these data are proven to be very valuable. Light curves have also been used for estimating ballistic coefficients for LEO RSOs, as proven in [37]. Additionally, light curves can be simulated as was performed by [38–41], which allows researchers to obtain large quantities of data on which to perform analyses.

Assessing an RSO's stability (spin rate or pointing stability) is useful as we identify the characteristics of RSOs through light curves even when we don't know the identity of the RSO we are observing. Once the stability of the RSO has been determined, we classify the RSO into various categories. Figure 2.1 illustrates some of the commonly accepted RSO classes as per [42]. These categories are not exhaustive or mutually exclusive, as RSOs can belong to multiple categories depending on the context.

There have been multiple studies on the use of ontology for RSO characterization and classification using the classes listed in Figure 2.1 [42–44]. The most basic classification is by object type. RSOs include satellites, rocket bodies, or other debris. Satellites are further classified by the presence or absence of an active control system which generates two subcategories: stable and tumbling satellites. Another common label for RSOs is the orbit type, which is determined from the altitude of the orbit. There are three main orbit types, LEO, Medium Earth Orbit (MEO), and high Earth orbit. A very common high earth orbit is GEO, which is typically placed as a separate category. GEO orbits are unique as they have an altitude of

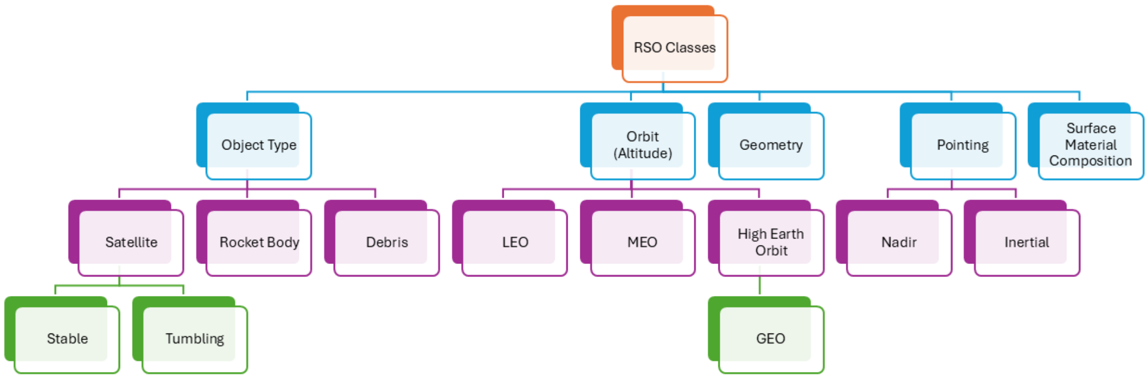


Figure 2.1: RSO classes.

35,768 km and an orbital period that equals Earth's rotation around itself. RSOs are also classified by their geometry and the direction they are pointing at.This is important as the shape of the RSO as perceived by the observer appears different depending on its attitude. This also affects studies of brightness, as the reflection from a solar panel is different from that of the bus. When considering observations of space objects, there are more categories that are useful in SSA, the most important of which is the material an RSO is made of, which impacts its reflection patterns.

2.1.2 Study Dataset

The dataset used in this research consists of light curves collected from the Ukrainian database and atlas of light curves [45]. This database contains a collection of light curves from 340 (LEO) RSOs with observations performed from 2012 to 2020. The dataset was gathered in Odessa using the KT-50 telescope which has an aperture diameter of 50 cm and is at an elevation of 56 m [46]. The light curves were collected from a KT-50 telescope in Odessa [45]. For each RSO, there are 1–343 observations taken at different time periods across a nine year time span. Most light curves have observation lengths of less than 15 min. The light curves are in the form of a matrix containing the modified Julian date and brightness for a specified United Nations outer space objects index catalog number that identifies the studied RSO [47]. For visualization purposes, the light curves for each RSO were plotted. Examples of light curves from RADARSAT-2 on 30November 2016, as a sample stable satellite, Jason-2 (OSTM) on 11 September 2020, showcasing a tumbling satellite, and CZ-4C on 14 July 2020, which is a rocket body, are depicted in Figures 2.2–2.4.

Both RADARSAT-2 and Jason-2 (OSTM) are boxwing satellites. Figure 2.2 has three distinct peaks which we assume to be caused by the antenna and solar panels. Figure 2.3 has many peaks, which indicates that the RSO completes more than one rotation during the observation. This is typical for tumbling satellites as they rotate much more quickly than stable satellites.

As for Figure 2.4, it shows a very different pattern where there are no distinct or wide peaks and the brightness constantly drops. By observing the light curve more closely, it appears that there are multiple local peaks despite the overall brightness decrease during the observation. This behavior is common among all rocket body light curves. The similarity in the brightness pattern could be attributed to the fact that the shape of the RSO is similar on multiple sides and there is no face that is significantly more reflective, as opposed to solar panels and antennas on satellites.

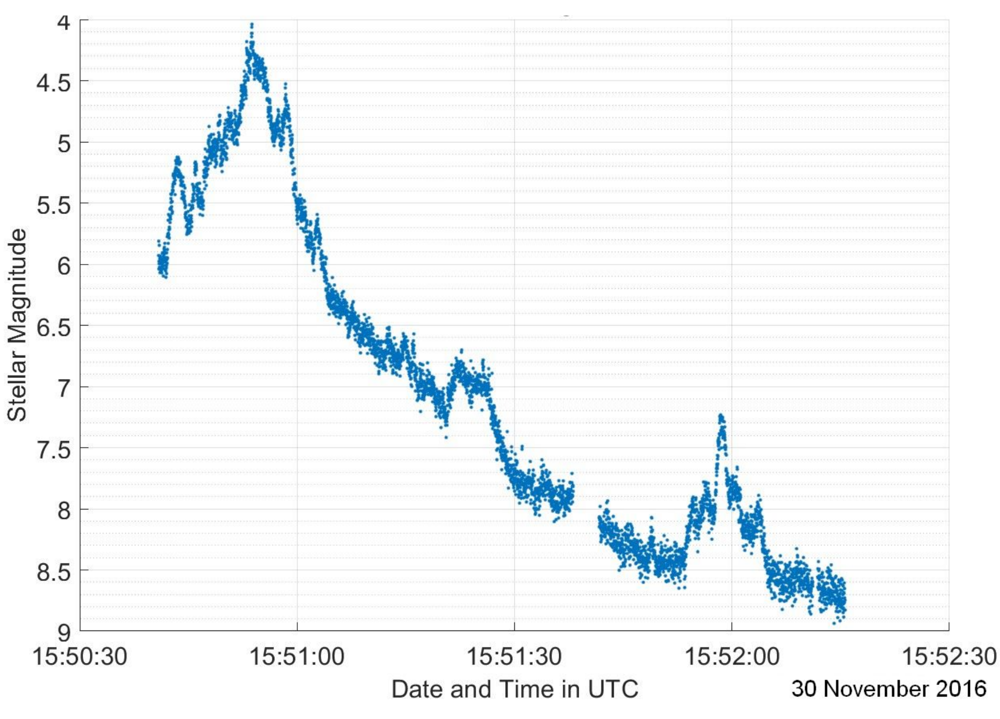


Figure 2.2: RADARSAT-2 light curve on 30 November 2016.

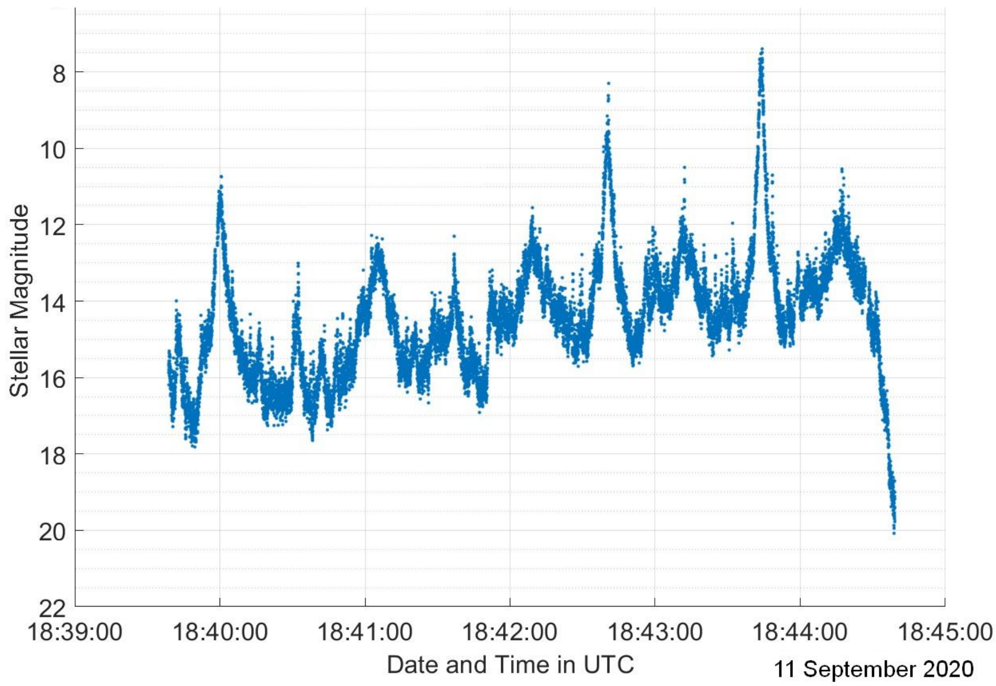


Figure 2.3: Jason-2 (OSTM) light curve on 11 September 2020.

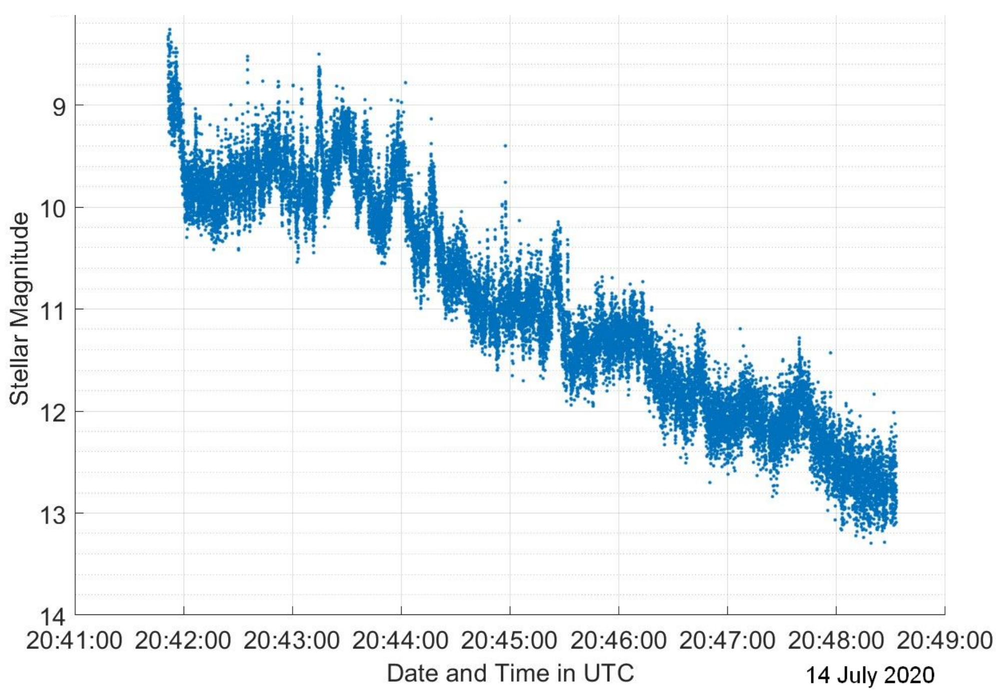


Figure 2.4: CZ-4C light curve on 14 July 2020.

It should be noted that these assumptions are applicable to LEO RSOs only, as GEO RSO light curves have not been analyzed in this study. Overall, we can assess that LEO light curves of stable satellites have very few peaks that are attributed to the surfaces with highest reflectivity. Tumbling satellite light curves have multiple distinct peaks or glints that have a repeated pattern. As for rocket bodies, their light curves have an almost linear brightness pattern with no clear peaks.

2.1.3 Literature Review on RSO Light Curve Classification

Many approaches are used for light curve classification, with ML being one of the most advanced and popular methods. The study in [48] provides a summary of different techniques of RSO classification. When researching different ML techniques for RSO light curve classification, there were two main types of studies. The first involves classifying light curves into specific known RSOs, these could be specific satellites, known satellite buses, or even certain rocket bodies. This type of research is demonstrated in [49, 50]. Multi-spectral simulated LEO light curves were classified in [49], which presents a unique approach in the literature for light curve classification. The main limitation with this form of classification is that it is used to identify a limited number of RSOs, and it is difficult to extend to the full RSO catalog as there are over 50,000 RSOs in the catalog.

The other form of light curve classification characterizes a variety of RSOs into different sets of categories such as shape, object type, or stability, to indicate if an RSO is actively controlled or not. This is the form of classification applied in this research. There are three aspects of this study that, when combined together, highlight the novelty of this work. These three points are: (1) LEO light curve analysis, (2) real light curve analysis, and (3) wavelet light curve analysis, which will be described in this section.

2.1.3.1 LEO Light Curve Analysis

Most studies on light curve classification are performed on objects in GEO due to the wide availability of measured light curves in addition to the availability of long duration observations of several hours. The works in [51–54] serve as examples of that. The work in [52] first attempts classification on a dataset of GEO light curves before applying the same algorithm on real light curves from a variety of orbits, including LEO and GEO. Studies on GEO light curves serve as a baseline for the design of LEO light curve classification algorithms, although the implemented strategies may not be the same due to the differences in the relative brightness, size, and motion of RSOs. Table 2.1 shows the difference between LEO and GEO light curves.

Table 2.1: Comparison of LEO and GEO light curves

Feature	LEO Light Curves	GEO Light Curves
Observation Duration	Short (typically a few minutes)	Long (typically a few hours)
Dataset Availability	Limited (only a handful of datasets exist)	Widely Available
Typical Size of Satellites	Variable	Large

For these reasons, LEO RSOs have not been studied as extensively as GEO light curves but, with the database and methodology used in this research, much more detailed analysis can be performed. The biggest difference between the dataset in this research and the ones used in GEO classification papers is that most observation durations are short, typically no more than a few minutes, with the longest observation being around 15 min in this study. There are only a handful of studies that discuss the classification of LEO light curves using neural networks, such as [55]. In [55], extra steps were taken for classification, including applying a test for periodicity and aliasing, aside from extracting the needed features using wavelet decomposition. In this work, we do not need to apply these extra steps and use the features from WST to train an ML algorithm with comparable accuracy. In [55], RSO characterization is performed using shape and spin rate of simulated light curves. However, in this study, we classify RSOs by object type and spin rate. The classifier sorts RSOs as stable satellites, tumbling satellites, and rocket bodies in this study, whereas the study in [55] classifies objects into stable objects, tumbling objects, or tumbling fragments. A tumbling object includes both satellites and rocket bodies and does not distinguish between the two.

2.1.3.2 Real Light Curve Analysis

There are many studies on light curve simulations, such as those in [38–41]. Several light curve classifiers are trained on simulated data with the expectation that it would closely match real light curves but, as indicated in [52], this is hardly the case. In [52], a Convolutional Neural Network (CNN) classifier initially designed for simulated light curves was trained on real light curves, which resulted in a 75.4% accuracy on real light curves and 97.83% on simulated light curves. Therefore, to build a classification algorithm that performs well on light curves collected in the field, real light curves must be used to train the algorithm. Although the study in [55] was on LEO light curves, analysis was only performed on simulated light curves in [55], whereas we use observational light curves.

2.1.3.3 Wavelet Light Curve Analysis

In order to extract the characteristics and features from light curves, we use wavelet transformation. Wavelet transformation is a frequency analysis method that is comparable to Fast Fourier Transformation (FFT). As opposed to FFT, which decomposes a signal into a collection of sinusoids, wavelet transformation breaks down the original signal into multiple wavelets, which are signals that are scaled and shifted with regards to a known signal or mother wavelet [56]. These wavelets are finite, as opposed to sinusoids that continue indefinitely. There are two main issues with the Fourier analysis listed in [57]: spectral leakage and Gibb's phenomenon. Spectral leakage is an issue where noise occurs during frequency analysis due to zero padding or other non-frequency related changes to the signal. This occurs because the finite inputs are then converted during Fourier analysis to a set of infinite sinusoids [57]. Gibb's phenomenon occurs when trying to fit a signal with abrupt changes and steep slopes to a set of sinusoids which are essentially smooth signals [58]. As a result, FFT does not work well for discontinuous signals and those that do not have enough samples for analysis. When analyzing observational light curves, wavelet analysis is used to address these shortcomings. In [55], a Discrete Wavelet Transformation (DWT) was used to obtain features from a light curve following the periodicity and aliasing test. In this study, we propose a different method of using wavelets.

To obtain a set of features from the light curves that are used for training ML algorithms, a technique called WST is used. WST is a technique that combines iterative convolution with wavelet transformation and low-pass filtering [59]. The equation for WST is provided in Equation 2.1 adapted from [60].

$$W_s = (s(t) \star \psi(t), s(t) \star \phi(t)) \tag{2.1}$$

In Equation 2.1, W_s represents the wavelet scattering coefficients, s(t) is the input signal which is the light curve data, $\psi(t)$ is the mother wavelet which is a morlet wavelet in this study due to its balance between time and frequency localization. $\phi(t)$ is the low pass filter and $\star$ denotes convolution. To apply WST, the signal is first convolved with the scaled mother wavelet. Then, the modulus of the convolved signal is applied to recover the information lost due to down sampling. Lastly, a low pass filter is applied to the resulting signal [59]. Using WST has multiple advantages, including stability in the face of signal time-warping deformations and noise, thus making it ideal for classification [60]. There have been multiple studies on applying WST to obtain features to classify ECG signals, which also consist of time series data, including [61–63]. In this study, we apply a similar technique to obtain features for LEO RSO light curve classification. The key contribution from this study is that measured LEO light curves are classified into three categories using only one major step for pre-processing: WST. This research tests the application of two different ML algorithms for classifying observational LEO light curves.

2.1.4 Artificial Intelligence (AI) Implementation of Time Series Classification

In this study, two different AI algorithms are applied for LEO light curve time series classification to determine the type of RSO. We first start by applying an SVM. Then, we perform the same classification with a neural network, specifically an LSTM algorithm. By studying two different types of AI algorithms, we determine the method that is best suited for light curve classification of RSOs.

2.1.4.1 SVM

SVM is commonly used for classification and regression for multiple reasons. Unlike neural networks, SVM is computationally efficient. SVM creates a hyperplane, essentially a multi-dimensional straight line, that separates the different classes. The defining feature of an SVM is the approach of selecting the hyperplane [64]. SVM adopts a concept called the maximum-margin hyperplane, where it selects the hyperplane that maximizes the distance between the different classes such that it is in the middle. The equation for SVM is provided in Equation 2.2 which is based on th equation in [65].

$$\min_{\mathbf{w},b} \frac{1}{2}|\mathbf{w}|^2 \quad \text{subject to } y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 \text{ for } i = 1:n \tag{2.2}$$

In Equation 2.2, w is the weight vector which contains the parts from WST that best separates the different classes, b is the bias term which permits the hyperplane to not neceesarily pass through the origin which makes the network more flexible, x_i is the network input, y_i is the output class for each light curve, i which is a value from one to the total number of light curves used in the network, n. The dot product is used to determine where each light curve is on the hyperplane and the constraint that the value is greater than one is a check that the light curve has been correctly classified. When the trends in the data are multi-dimensional, a kernel is used which adds extra dimensions as needed for improved performance. SVM is also applied for multi-class classification where there are more than two output cases. An in-depth analysis of SVMs is available in [66]. An SVM has been used in astronomical research to classify light curves of stars as in [67], so it should also be sufficient for LEO RSO light curve classification.

2.1.4.2 LSTM

LSTM is a type of Recurrent Neural Network (RNN) that resolves the vanishing and exploding gradient issue typically associated with RNN [68]. LSTM consists of three gates or parts. The first determines the signal at the current step, the second determines the signal at the previous step, and the last is the "forget" gate which determines how much of the previous signal to consider in the current signal. Combining previous and current steps makes LSTM a robust tool for analyzing time series data [69].

LSTM has been widely used for time series classification, such as in [70, 71]. It has also been applied to similar research problems, such as star classification [72]. For LEO RSO light curve classification, LSTM has been used in combination with a Hidden Markov Model (HMM) in [55] for simulated light curve classification. To train an LSTM, there are multiple hyperparameters that must be carefully selected to obtain the best performance. There are a wide range of hyperparameter values to choose from, which makes the task even more daunting. There are multiple methods to choose the best hyperparameter values, but the method implemented in this study is Bayesian optimization.

2.1.5 Bayesian Optimization

Bayesian optimization is applied to obtain the hyperparameter values that result in the best algorithm performance and is especially useful when the function we need to minimize is overly complex as it considers the function as a black box. The goal of Bayesian optimization is to find the set of requested parameters that minimizes the value of an unknown objective function [73]. Bayesian optimization is considered a Gaussian process where the value of the next iteration is based on the current iteration [74]. Bayesian optimization is also much less computationally expensive than an exhaustive sweep over all potential options. A more extensive analysis of using Bayesian optimization for choosing hyperparameters is in [75, 76].

Using Bayesian optimization, the optimized hyperparameter values are 500 hidden units, 220 epochs and a mini-batch size of 50. The number of hidden units impacts the size of the network and the model's capacity to learn from the data. The number of epochs refers to the number of iterations the LSTM uses for training, and the mini-batch size is the size of the subset used for training.

2.2 Generating Training Features from Light Curves

This section explains the steps taken to obtain the training features from the light curves used in this study.

2.2.1 Data Formatting and Extraction

Prior to applying WST, the data from [45] required some formatting to be more intuitive. In the original data, there is one file for each RSO with numerous observations taken at different dates listed sequentially. For this research, we separated each observation into a separate input file. This makes plotting and subsequent analysis much more intuitive as different RSOs are detected at different times.

Aside from the multiple observations per file, there were two main issues in the original data. Each observation is typically sorted in order of increasing time but there are some that stray from this norm. There are duplicate measurements that are unsorted. The lack of sorting is problematic when performing any form of time series analysis, so the pre-processing code contains an extra feature to sort the data in each observation. Another, more serious, concern is that there are some observations with a constant magnitude of 36 most likely due to missing samples. This is a critical issue as it greatly skews the results of subsequent analyses. This was remedied by first locating and then removing the anomalous observations during pre-processing. The light curves described in this study have all been formatted by this method. The next step involves extracting the labels. The dataset contains light curves accompanied by the catalog number which indicates the RSO's identity. To store the needed labels for a large collection of RSOs, we developed a script that would obtain the needed information from several websites, such as Gunter's Space Page and N2YO, and store the label in an Excel sheet.

Given the large amount of data needed for this study, typical methods of loading variables into memory are very computationally expensive, so we developed a custom datastore that records the location of each light curve file alongside its label. The label contains the class of the RSO, which is one of three options: stable satellite, tumbling satellite, or rocket body. This allows us to perform the needed transformations on the data while out of memory for most of the code and only reading the data into memory at the last step when obtaining the features. More information on datastores is found in [77, 78]. The training and test data in this study are partitioned per light curve and not per type of object.

2.2.2 Feature Extraction

As mentioned in Section 2.1.3.3, WST is applied to the light curves to generate a set of training features to be used for classification. In order to create the scattering network, the characteristics of the inputs need to be provided. This includes specifying the signal length, invariance scale, and sampling frequency. The signal length represents the number of samples available. The invariance scale refers to the limit of the low pass filter beyond which the signal is reduced to zero. The sampling frequency is the rate of capturing data, which has been determined as approximately 48 Hz for the dataset used in this study.

After creating the scattering network, the scattering feature matrix is obtained. To generate valid inputs for classification, a constant signal length must be chosen. Given that the light curves have varying durations, a single signal length must be chosen, and the remaining data must be padded or truncated. There are many ways to choose the signal length. In this study, the value that maximizes the training accuracy while having enough training inputs was chosen. If a light curve has been collected for longer than the specified duration, it is truncated. As for shorter duration light curves, there are two possible ways for them to be pre-processed. The first method simply eliminates all light curves with durations shorter than the specified duration. Alternatively, the shorter signal is padded with zeros to have the applicable number of samples. For this study, the shorter duration light curves have been removed but we have included the option for padding the signal in the algorithm if it is deemed necessary for future studies. Additionally, if required by the dataset, normalization is applied but, for this study, it made no impact due to the consistency of the inputs. The features consist of a 3D matrix with the first dimension indicating the number of scattering paths, the second referring to the number of scattering coefficients, and the third dimension referring to the total number of light curves.

2.2.3 Input Data Visualization

To better visualize the different classes of light curves, we implemented a t-distributed Stochastic Neighbor Embedding (t-SNE), which is a statistical technique for converting multi-dimensional data into two or three dimensions, which is easier to visualize [79]. Similarly structured data appear to be clustered together, which differentiates between the different object classes. Unlike Principal Component Analysis (PCA), which is a linear technique, t-SNE is a non-linear technique that preserves the local trends in the data. The equation for t-SNE is in Equation 2.3 and is obtained from [79].

$$t\text{-}SNE = min(\sum_{i,j} P_{ij} \log \frac{P_{ij}}{Q_{ij}}) \tag{2.3}$$

In Equation 2.3, P_{ij} is the pairwise similarity between two points in the original data space and Q_{ij} is the pairwise similarity in the lower 3D space. There are a few parameters that are used in t-SNE. The perplexity is a value that represents how many nearest neighbors to consider in calculating the pairwise similarity and its value is set as 30 in this study. Additionally, there are multiple distance metrics that can be used to compute the t-SNE, but in this study we implemented the Euclidean and Chebychev distance equations on the light curves of the three different classes to generate a 3D embedding of the input data shown in Figures 2.5 and 2.6. Figures 2.5 and 2.6 reveal different clusters in the data. For example, most stable satellites are clustered together in the Euclidean embedding. This does not hold for the Chebychev distance calculation though, where stable satellites and rocket bodies appear to cluster in a similar region. By applying t-SNE, we can see some separation among the different classes but it is not very pronounced. This makes it difficult for classification. After applying WST to match the input to the different algorithms, the relationship between the different coefficients now has a different trend, which is shown in Figures 2.7 and 2.8.

After applying t-SNE on the WST features, we can notice a closer similarity between the Euclidean and Chebychev distance approaches. This is purely an artifact of the data though, and does not necessarily indicate an underlying trend in the data. The classes are still not completely split into individual clusters, but we can notice that certain cluster regions now appear for the three different classes. Given that the classes are not appearing in tightly grouped clusters, the light curves will not be simple to classify and there may be some outliers in the classification results.

2.3 Light Curve Classification

To classify light curves, we first implemented an SVM with two different methods of preparing the inputs to study the difference in performance. Then, we trained an LSTM RNN to assess the difference between a classical ML approach and a neural network. To compare the results, we computed some performance metrics such as accuracy, precision, recall, and F1 score. Since we have a multi-class

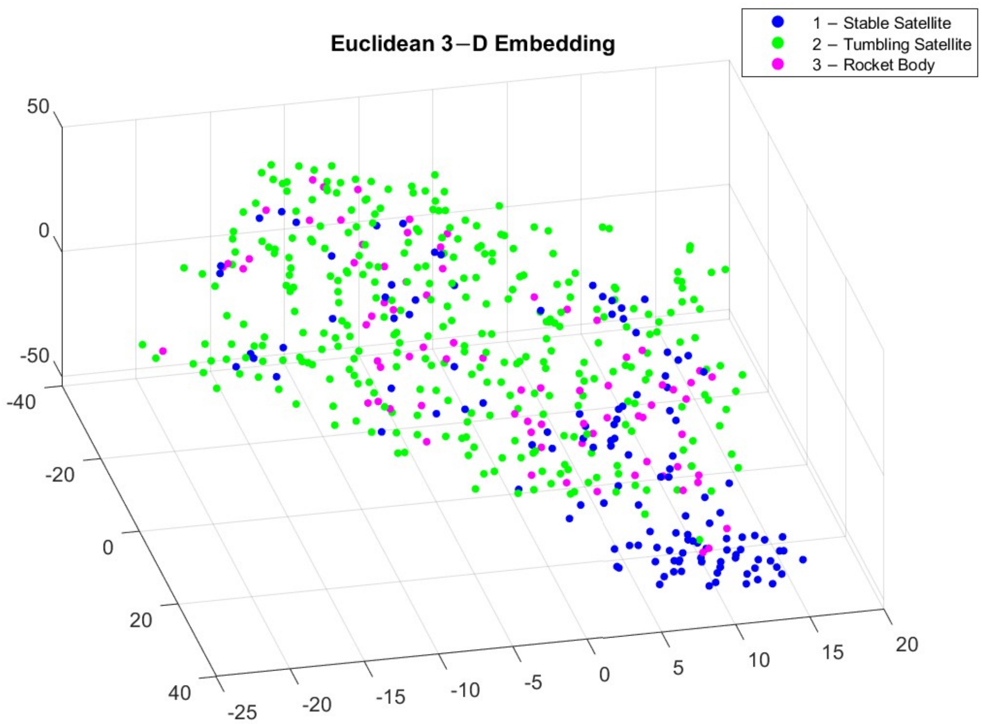


Figure 2.5: 3D embedding of light curves using Euclidean distance.

-

classification problem with a different number of elements per class, weighted averaging was used. The definition and equations for each of these terms are defined in detail in [80, 81].

2.3.1 SVM Results

When implementing an SVM, we utilized two different approaches for generating the model inputs. We attempted a mean features approach and a window-based approach. For both types of inputs, we defined the same training parameters. To obtain higher accuracy results, we implemented a second order polynomial kernel as opposed to using a linear function. The next factor to choose was the type of multi-class classifier. When there are more than two output classes, there are two main ways to specify how the computation happens. The first approach is a one vs. all classification, where each class is compared with all other classes [82]. The second approach, called one vs. one, breaks the multi-class classification problem into a collection of binary classifications, where each class is compared to one class at a time forming pairs. A deeper comparison of the two techniques is in [83]. For this study, we

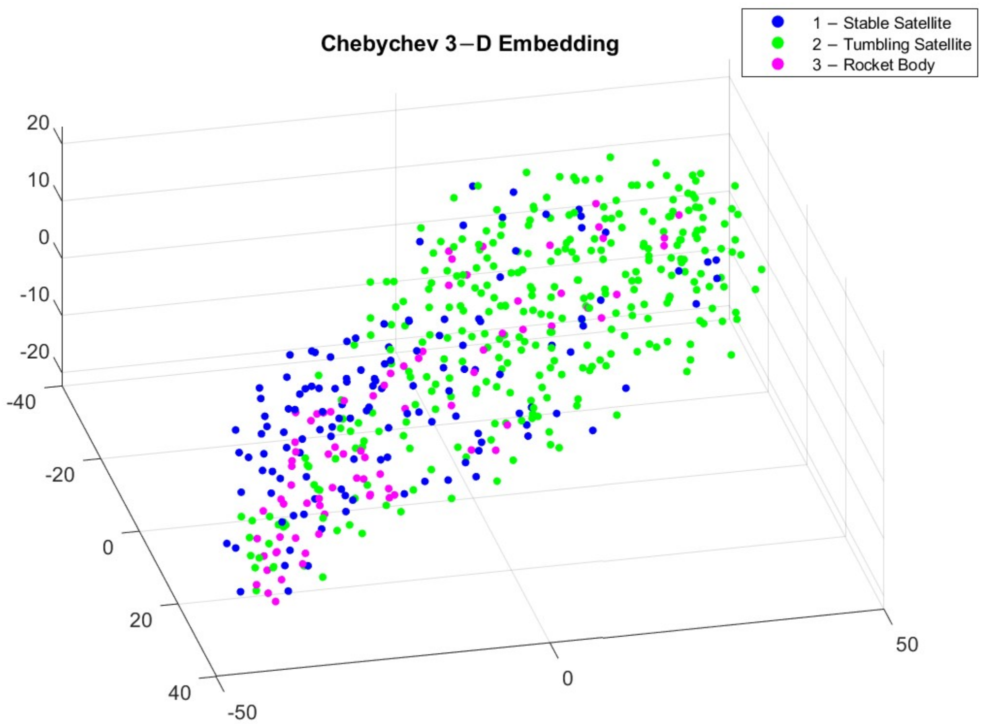


Figure 2.6: 3D embedding of light curves using Chebychev distance.

showcase the results of using a one vs. one classification approach as we noticed minimal change on the testing accuracy from a one vs. all approach.

2.3.1.1 Window Based Approach

The first method of preparing inputs for SVM we attempted is by using a window-based approach. To perform this, we split each scattering time window as a separate input to the model and trained the SVM. When computing the performance metrics, we combined the signals back to the initial input size and took the label featured most for each RSO as the predicted label. The overall accuracy of this method was 60%, which is unacceptable for classification. Table 2.2 summarizes the performance metrics. The performance of the algorithm is poor specifically when considering precision, recall, specificity, and F1 score.

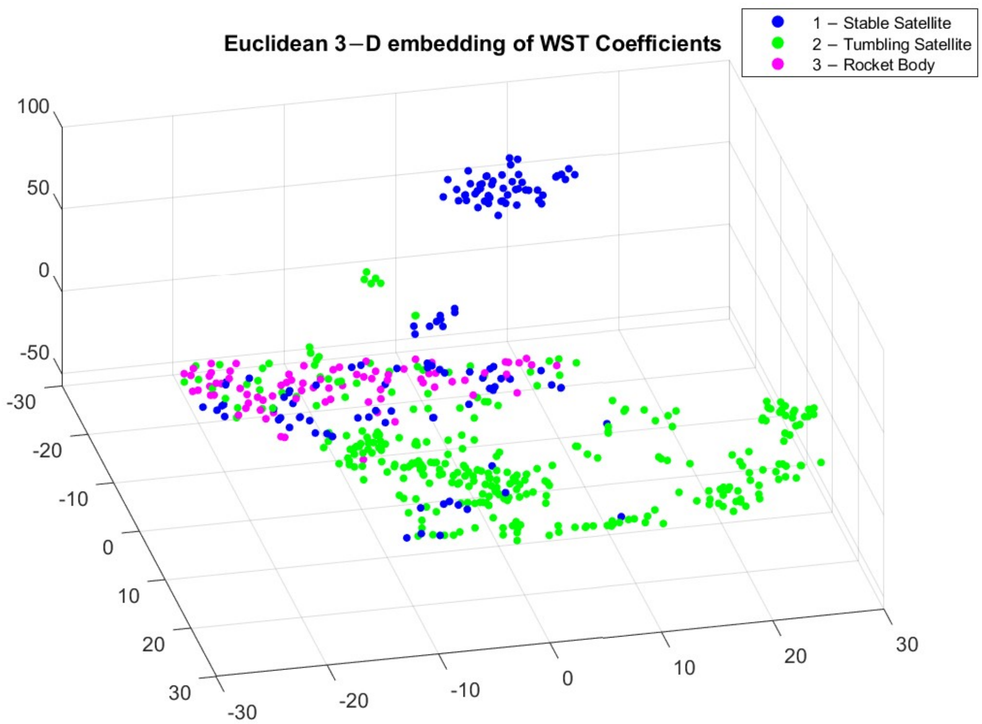


Figure 2.7: 3D embedding of WST coefficients using Euclidean distance.

Table 2.2: Performance metrics of different light curve classifiers.

Method	Accuracy	Precision	Recall	Specificity	F1 Score
SVM Window-Based Approach	60%	40%	48%	46%	40%
SVM Mean Features	87%	86%	82%	84%	81%
LSTM	92%	90%	89%	95%	89%

2.3.1.2 Mean Features

The second type of feature pre-processing involves using the mean features from WST as inputs to the SVM. This allows us to obtain a 2D set of features from a 3D array. The average of the columns, which consist of the second dimension of the matrix, is taken such that an $m \times n \times o$ sized-matrix becomes an m by o sized array. By reducing the dimensional of the input, we reduce computation time significantly. The resulting test accuracy is 87%.

The overall performance of the classifier when using mean features has improved significantly as compared to the window-based approach (see Table 2.2), and the algorithm runs about 86% quicker

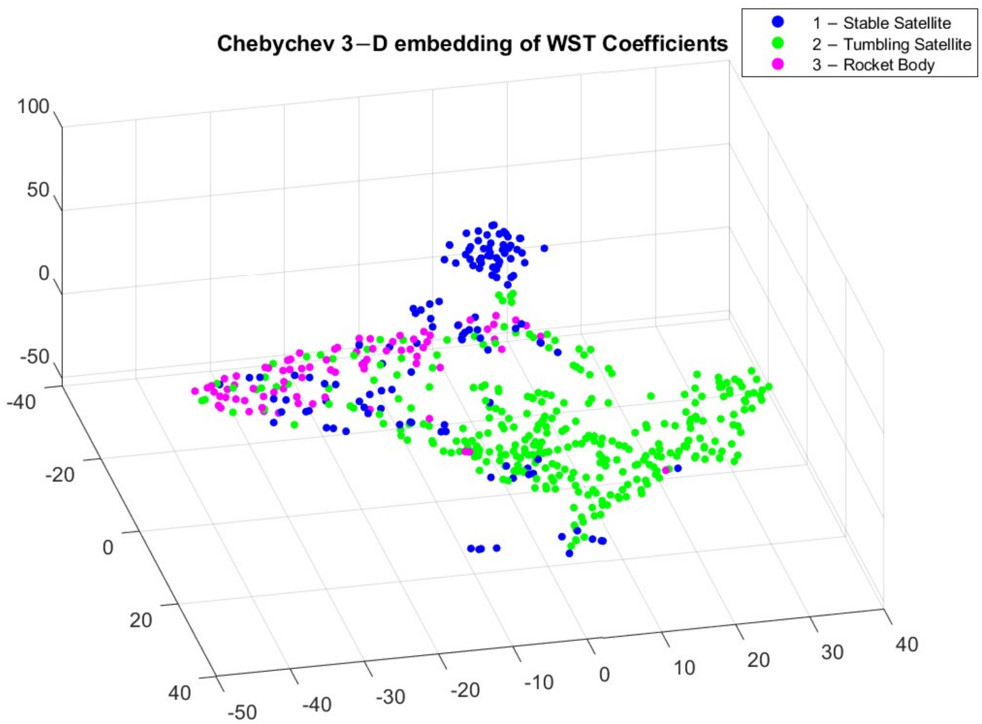


Figure 2.8: 3D embedding of WST coefficients using Chebychev distance.

as it only takes seven seconds to train the network. Given that the values of all metrics are relatively similar to each other, we conclude that this approach results in more consistent classification regarding False Positives (FP) and False Negatives (FN).

2.3.2 LSTM Implementation

To study whether a neural network performs better LEO light curve classification as compared to a simpler approach like SVM, we trained an LSTM. This method was chosen specifically due to its wide use in the literature for multi-class classification. As opposed to SVM, which requires a few options to choose from before training, neural networks require the user to supply many hyperparameters, and the sheer number of potential options require some form of optimization technique to determine the best values to choose from. As a result, we implemented Bayesian optimization to select the best values for the hyperparameters.

2.3.2.1 Applying Bayesian Optimization

There are many hyperparameters that we could optimize but, for this analysis, we chose to prioritize the number of hidden units, the maximum number of epochs, and the mini batch size, as well as the initial learning rate. The number of hidden or recurrent units refers to the number of internal layers within the LSTM network. As indicated in [84], if this number is too large, the network overfits the data quickly, which deteriorates the performance on a test set, but if the number is too small, then the network does not store enough details to efficiently solve the problem. The number of epochs refers to the number of times the model passes through the complete training set. Similar to the number of hidden units, too small a value misses essential features of the data, while too large a value risks overfitting [85]. Mini batch size refers to the number of inputs to use for each iteration. Thus, each epoch has a collection of batches. Finally, the learning rate indicates the change in weight of the model. A very small value takes too long to train and a very large value risks overshooting and diverging from the expected result.

2.3.2.2 LSTM Results

After obtaining the hyperparameter values using Bayesian optimization, which took over a week to optimize 30 epochs, the LSTM network was trained, which resulted in 92% accuracy. Additional performance metrics are provided in Table 2.2. Training the LSTM was completed in 3.5 min, which is 30 times longer than SVM mean features. Although this algorithm is slower than the mean features SVM approach, LSTM has the best performance as compared to both SVM approaches with regards to all five performance metrics. This means that, although simpler solutions like SVM work relatively well for light curve classification, higher accuracy demands more complex methods such as deep learning with LSTM used as an example in this study.

2.4 Conclusions

In this study, we implemented a novel approach for LEO RSO classification of observational light curves. We used WST to extract features from LEO light curves, which was then applied to different AI algorithms to classify RSOs by object type (stable satellite, tumbling satellite, and rocket body). By visually observing the different light curves, we determined some differences between stable satellites, tumbling satellites, and rocket bodies, but these differences are complex, which make it difficult to apply analytical classification. This is further proven by applying t-SNE to the input light curves as well as the WST coefficients, as the clusters are closely grouped together. Applying t-SNE on the WST coefficients adds more distinction between the classes, but the result is not pronounced enough to warrant simpler classification techniques feasible. We applied a non-deep learning technique (SVM) as well as a deep-learning approach (LSTM) to compare classification performance.

We used the scattering network features to train both the SVM and LSTM for comparison. For the SVM, we first tested a window-based approach where the scattering network was split by each individual time window. Although we expected this method to be satisfactory for classification, the resulting accuracy was only 60%. The low accuracy is most likely due to the lack of sufficient features in each time window and the resulting large amount of training data due to splitting each time window as a separate input. SVM does not work well for very large datasets, and it seems to be the main cause of the poor performance. The second approach to SVM used the mean features of the scattering network as input parameters. Although we removed one dimension from the data which decreased computation time and simplified the inputs, the accuracy has increased significantly to 87%. This also proves that the large dataset was the reason for the time window approach failing. Finally, we trained an LSTM on the features from the scattering network as is. A major complexity of neural networks is determining the ideal value of the hyperparameters that result in the best performance and, in this case, Bayesian optimization was used. Using the hyperparameters from Bayesian optimization resulted in the highest accuracy out of all approaches, 92%. This is significant to the SSA community as this study provides the capability of classifying observational LEO light curves into stable satellites, tumbling satellites, and rocket bodies.

Future work includes implementing (1) a light curve classification algorithm on datasets of observational light curves taken from a space-borne camera, as well as looking at (2) the improvement of classification accuracy for each class. For space-borne datasets, we are currently examining FAI and NEOSSat images. FAI is an instrument onboard the CASSIOPE satellite [86]. As indicated by its name, the original purpose of FAI was to study auroras, but the camera specifications work for RSO observation as well. The camera specifications are comparable to a star tracker in that it has a low resolution of 256x256. NEOSSat is a Canadian satellite that targets asteroids [87]. Ground-based images are captured from a fixed observer, but space-based images are captured from a moving, and sometimes spinning, satellite.

Therefore, space-based images are much more complicated for light curve extraction and classification. Another aspect that impacts the generation of light curves is the method of capturing images. An observer can either point at a fixed location and capture images, which we refer to as stare mode, or track RSOs as they pass through the FOV, referred to as tracking mode. Light curves have traditionally been captured in tracking mode, but we are interested in attempting stare mode light curve extraction due to the wide availability of such images.

Chapter 3

LEO RSO Light Curve Classification using Barlow Twins

Abstract

The number of satellites launched into space is increasing by the day especially with the proliferation of large constellations. As a result, there is a growing need to track and classify RSOs such as satellites, rocket bodies, and debris to keep our satellites safe and clear from collisions. One approach to understanding the type of RSOs is by using their light curves which are time series of brightness measurements. Although most RSOs are in LEO, few LEO RSO light curves have been studied due to the short measurements and lack of sufficient data. Despite these challenges, it is very beneficial for space operators to have an understanding of how RSOs near their satellite move. In this study, we demonstrate a technique for LEO RSO light curve classification using a self-supervised ML technique referred to as Barlow Twins. We tested this algorithm with light curves that are two, three, and five minutes long and obtained accuracies of 75%, 85%, and 97% respectively when classifying light curves into stable satellites, tumbling satellites, and rocket bodies. By distinguishing these classes, space operators can prioritize avoidance maneuvers for critical targets to protect the orbit around operational satellites.

3.1 Introduction

There is a rapid increase in the number of satellites launched into space. As a result, the orbit around the Earth is getting more congested. As a result, the chance of collisions occurring between different RSOs is increasing [88]. RSOs consist of objects in space including satellites, rocket bodies, and debris. The risk is especially large for RSOs in LEO where the number of launched and planned constellations is growing. The increase in the number of RSOs in space necessitates the actions of detecting, tracking, characterizing, and identifying RSOs, which are encompassed in a field known as SSA. SSA studies are typically performed using astrometric analysis of starfield images containing RSOs. Although this is useful for SSA, this approach alone does not capture all possible information about the RSO such as its spin rate. To better understand a satellite's shape and its spin rate, the object's brightness can be observed over time. This brightness data is known as a light curve. RSO light curves allow us to understand characteristics of RSOs without using images. Given that we only need brightness values, there is no need to downlink entire images from RSOs in orbit. Alternatively, ground-based telescopes are also used to measure RSO brightness. Light curves allow us to extract information about an object such as the type of RSO (whether it is a satellite, rocket body, or debris), the attitude or orientation of the RSO, and the shape of the RSO as well as the material it is made of. Understanding what RSOs are in the FOV of an observer and how they move allow operators of the observer satellite to better perform maneuvers to prevent collisions and keep our orbits safe. Specifically, knowing the type of object allows operators to prioritize tracking and maneuvering decisions to avoid potential collisions. For example, unstable objects with irregular spin patterns or rocket bodies with unpredictable trajectories have a higher risk and should be targeted for more extensive monitoring and collision avoidance actions. By focusing on riskier and non-stable objects, operators can optimize their resources to maintain the safety of their RSOs of interest. In the field of astronomy, light curves of stars and asteroids have been researched extensively for decades but not all the techniques apply to RSO light curves given that the light curves studied in astronomy tend to be much longer than the ones we obtain for RSOs.

There are many factors that impact light curves. Some are local to a certain region of the light curve measurement, and others are global to the whole RSO. Global factors include the RSO's phase angle and spin rate, while local factors include the shape and material of the RSO particularly when considering surfaces that have antennas or solar panels which appear brighter to an observer. In the field of RSO light curve classification, most research advances have been on GEO RSOs that are at an altitude of around 36000 km as opposed to LEO RSOs with altitudes less than 1000 km. As noted by [51, 53], the increase in observation systems built from Commercial Off-The-Shelf (COTS) components expands the number of GEO observations performed and demonstrates the ability to collect large datasets for full GEO belt observations. GEO RSOs are also capable of being observed for long periods at a constant elevation angle which results in long light curves as well. On the other hand, LEO light curves are much shorter than GEO light curves, on the order of a few minutes as compared to a few hours. Therefore, LEO light curves are more challenging to research. Even though there are challenges, studying LEO light curves is of great importance as 84% of RSOs are in LEO [89]. Due to the small number of GEO satellites, researching GEO RSOs alone severely limits our understanding of most RSOs in space. This makes LEO light curve studies extremely beneficial to the space research and operations communities. The main objective of this paper is to improve an existing algorithm for LEO light curve classification by RSO type and stability where the algorithm classifies RSOs into stable satellites, tumbling satellites, and rocket bodies.

Light curve classification on LEO objects have been limited to simulated light curves such as in [55] or to longer measured light curves including our own study from 2023 [1] where we examined measured LEO light curves with five minutes of observation period. Most LEO light curves are much shorter than five minutes in length due to the severely limited viewing angle. In the dataset used in this study as well as the 2023 study [45], over 90% of the light curves were obtained with less than five minutes of observation. A histogram for the lengths of the light curves in our study is in Figure 3.1. Prior to our implementation for light curve classification, the light curve dataset has been used to study the rotation rate of RSOs [46]. In our previous work [1], we implemented SVM and LSTM for light curve classification [1]. The best performance of the algorithm yielded an accuracy of 92%. Another aspect to note regarding our previous algorithm is that instead of using light curves directly, we used WST coefficients of the light curves as the input to the ML algorithms. While WST allows us to extract suitable features for accurate classification, using the original dataset without feature extraction is preferable with complex networks that are capable of learning features directly from the raw data. Additionally, using the original data allows for the preservation of information that would be discarded when extracting features. Furthermore, using the original dataset simplifies the ML model and reduces processing time. Once the network has been trained, the extra step of obtaining features for the test set is removed which improves the end user's understanding of the network and simplifies the process of using the algorithm for future test cases.

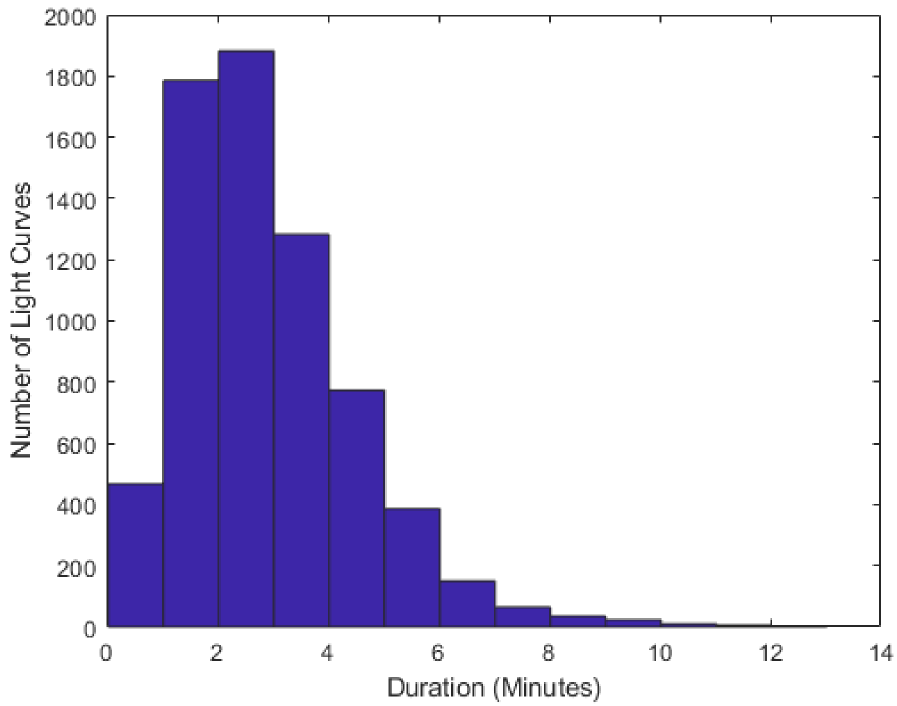


Figure 3.1: Histogram of Light Curve Durations.

The Barlow Twins presented in this study addresses both the short light curve and WST issues when classifying LEO light curves to obtain a more efficient and comprehensive algorithm.

As we noted in [1], there is growing interest in RSO light curve classification using ML and AI. However, most studies thus far are limited to either simulated light curves or GEO objects. In this paper, we review the ML approaches used in light curve classification and introduce the application of the Barlow Twins method, originally developed for image classification in [90], to the classification of LEO light curve data.

3.2 Background - Approaches to light curve classification

3.2.1 ML Methods

The most common application of ML in light curve classification is in the astronomy field. ML classification of astronomical light curves of stars and asteroids have been extensively studied in many works. In [91], real light curves of stars were classified to determine whether there is a transiting exoplanet in the light curve or not. This study was the first to demonstrate the use of real light curves using ML classification. A Random Forest (RF) was determined to have the best performance with an accuracy of 74.71%. In [92], a CNN was used for astronomical transient event detection and was determined to outperform an RF. The resulting accuracy of the CNN is 99.45%. Research performed in [93] demonstrates an automated classification algorithm using boosted decision trees for a variety of astronomical objects including stars and supernovae. The classification accuracy varied enormously amongst the different classes as astronomical objects were classified with an 81% - 100% accuracy while supernova classes had accuracy values of 33% - 71%. Countless other research on astronomical light curve classification has been performed such as in [94–97]. With the large number of potential astronomical light curve classification labels, a direct comparison of these different studies is not possible as each study performs a different classification task with a different dataset. Nevertheless, these studies show that there are many successful attempts to astronomical light curve classification.

Astronomical light curves capture changes in brightness over time, typically caused by sunlight reflected off asteroids or planets or by light emitted by distant stars and galaxies located many light-years away. As a result, light curves could be several hours long. Observed RSOs, on the other hand, are at much shorter distances from the observer, whether that is a ground-station or another RSO. As a result, RSO's pass by the observer much faster and the measurements are shorter than astronomical light curves.

Most advances in RSO light curve classification have been made for GEO RSOs. In [40], simulated GEO light curves were classified using a CNN. In [51], a multi-step classification approach was applied using an RF network and a HMM. The first classification step determines whether an object is stable or tumbling and the second step classifies tumbling bodies to rocket bodies or tumbling satellites. In [52], SVM, CNN, and Bagged Ensemble approaches were tested for simulated GEO light curves as well as measured light curves of RSOs in LEO and GEO. The Bagged Ensemble approach in [52] achieved a 99.5% accuracy on the simulated dataset but 63.5% accuracy on the real dataset. The large drop in accuracy when using real light curves highlights the challenges of generalizing models trained on simulated data alone when dealing with complex data such as RSO light curves. Using measured light curves alone would yield more realistic results that will allow the network to be used effectively by satellite operators.

Given the importance of LEO RSOs, it is critical to study LEO light curves in addition to GEO. The major difference between LEO and GEO light curves that prevent us from directly implementing past solutions is the much shorter duration of LEO light curves that are only a few minutes long as compared to GEO light curves. There are some emerging studies on LEO light curve classification such as [55] where LSTM and HMM were used for simulated LEO light curve classification. To our knowledge, our previous work, [1], presented the first approach to real LEO RSO light curve classification. In [1], we used WST to obtain the inputs to the network. We then used Bayesian Optimization to determine the optimal hyperparameter values to train the network. We tested both SVM and LSTM architectures. The network that yielded the best performance was an LSTM network that achieved over 90% accuracy.

With regards to ML light curve classification, there are some similarities across the different approaches indicated earlier. First, most ML algorithms rely on features that are used as inputs to the ML algorithm as opposed to the light curve directly. These features include the minimum, maximum, and average brightness, the linearity of the light curve, and the slope. In addition to counting certain statistics from light curves and using them as features, coefficients determined from light curves are used as well. In [55], DWT is applied to determine the coefficients used for training and in our previous work, [1], WST was used. In our experimentation, we deemed the iterative convolutions and low-pass filtering incorporated within WST to be a better fit than DWT for classifying measured LEO light curves. One challenge is that WST requires defining parameters for efficient feature extraction. In addition to the sampling frequency, the invariance scale, which represents the time scale of the low pass filter, is another input. Considering that WST features are not easily interpreted on their own, re-training the full network when changing these parameters is needed to optimize these values. Therefore, extracting features from light curves involves additional processing. If this step can be removed, a more efficient algorithm can be designed. In astronomical light curve classification, [95] uses density based parameters of the light curve using the distance between measurements as opposed to static features. In this work, we demonstrate an improved classification algorithm that uses augmented light curves directly in the ML algorithm as opposed to features. Once the network has been trained, testing can be performed on the direct light curves alone without any augmentation.

Another aspect to consider when choosing an ML algorithm is the type of learning needed for the model. There are supervised, unsupervised, semi-supervised, and reinforcement learning approaches. Most light curve classification models utilize supervised models alone while some such as [51, 55] combine a supervised model with an unsupervised HMM to assist with the classification. Our initial work in

[1] used supervised learning. In this study, we use a self-supervised ML algorithm, namely Barlow Twins, followed by a softmax layer to perform the classification. The light curves used in this study are manually labelled. As a result, adding a softmax layer after the main network has been trained prevents the model from relying too heavily on potentially mislabeled data.

3.2.2 Analytical Methods

In comparison, there are non-ML based methods that have been successfully applied to characterize light curves. In [98], FFT, classical periodogram, Lomb-Scargle periodogram, Phase Dispersion Minimzation (PDM) and other analytical techniques were used for sidereal rotation period determination. PDM and periodograms were deemed to perform the best. These techniques were applied to measured astronomical light curves as well as simulated LEO light curves. Lomb-Scargle periodogram is advantageous in analyzing data that is difficult to study with FFT such as the case of an unevenly sampled dataset. PDM starts by assuming different periods and calculates the ratio of the data conforming to those periods. The most likely period is one that minimizes the error ratio. The light curves are then folded over or resampled at these periods to form more uniform light curves as compared to the original. The analysis was successful for the astronomical light curves but was not as reliable for the analysis of simulated LEO light curves. This issue can be attributed to inconsistencies in simulating LEO light curves. In [36], measured light curves of spherical and cylindrical objects were characterized using a custom phase function. In our research, prior to implementing ML techniques, we attempted LEO light curve classification via analytical methods such as FFT, Lomb-Scargle, and PDM. These approaches are used for spin rate extraction. No unique trend from the stable satellites, tumbling satellites, and rocket bodies were determined from the analytical analysis. As a result we concluded that the real light curves are too short to discern the spin rate using these methods. This was determined by looking at the small "periods" calculated by PDM in addition to the lack of significant spectral components in the periodograms. Additionally, analytical approaches work well when tuned for the dataset where the shape or other features are well known which is not the case in the measured LEO light curve study. While useful for small datasets where the shape of the object is known ahead of time and the light curves are long enough, analytical analysis is not sufficient for accurate batch light curve classification when given a large dataset of light curves where the RSO's shape is not always known and the observations are a few minutes long.

Table 3.1: Distribution of RSO Classes in the Light Curve Dataset.

RSO Class	Percentage of Dataset
Stable Satellites	28%
Tumbling Satellites	37%
Unknown Satellites	14%
Rocket Bodies	17%
Debris	4%

3.3 Dataset

The dataset used in this study, as well as our previous work on light curve classification, consists of light curves from the Ukrainian Database and Atlas of light curves of artificial space objects which contains more than 6,800 LEO light curves of over 300 RSOs [45]. The dataset is sampled at a frequency of around 50 Hz so the data points are approximately 20 milliseconds apart from each other. In this dataset, many RSOs have multiple light curves captured at different time periods. Another notable aspect of the dataset is that the duration of each light curve is different. The last thing to note is that the original dataset consists of the light curves (brightness and time values) and the identity of the RSO. The RSOs were manually labelled into the classification categories: stable satellites, tumbling satellites, and rocket bodies. The type of object (satellite, rocket body, or debris) is easily determined using the RSO's catalog number which identifies the RSO. Using this information, an algorithm was developed to scrape the internet to obtain the label. The more challenging task was to determine whether the satellite is stable or tumbling. This required a more manual approach of searching for the satellite's status. There are some satellites without publicly available status and those were simply labelled as "Unknown Satellites". It should be noted that acquiring and labelling the dataset was performed when working on the original classifier in [1]. The percentage of each label in the dataset is in Table 3.1.

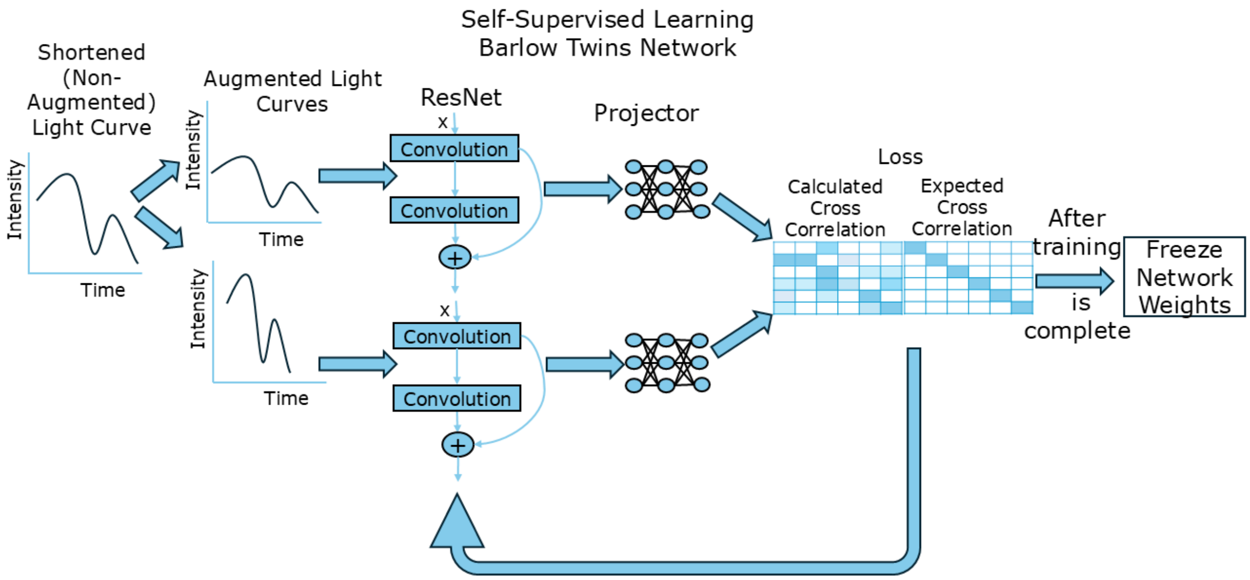
3.4 Methodology

A Barlow Twins network first described in [90] is used for light curve classification. This is a type of self-supervised learning algorithm that uses redundancy reduction. This architecture consists of two random augmentations of each input where the cross-correlation matrix from the embeddings of these random augmentations, or Twins, is as close to the identity matrix as possible. The Barlow Twins network was originally developed for image classification, but we are applying it in this study to time series data, namely light curves. There are multiple reasons for considering Barlow Twins. Given that Barlow Twins is a self-supervised network, the algorithm is less reliant on the manually assigned labels such that potential mislabelling does not alter the way the Barlow Twins algorithm learns from the data. To achieve classification, we added a softmax layer after the Barlow Twins implementation which relies on the labels, but this is only added after the network has been trained without the labels. This approach prevents the network from learning skewed representations that could be a result of inaccuracies in manual labelling, particularly between stable and tumbling satellites. Another factor is that Barlow Twins is a flexible approach that allows the use of different types of encoders without impacting the implementation of the overall architecture. A Barlow Twins network is implemented with encodings from Residual Networks (ResNets), Transformers, Variational Auto Encoders (VAEs), or other architectures as deemed necessary by the study being performed. This allows for experimentation with different encoders while keeping the main structure of the network intact. Additionally, by using two random augmentations per light curve, the algorithm learns to be robust against minor changes that do not alter the characteristics of the object being observed.

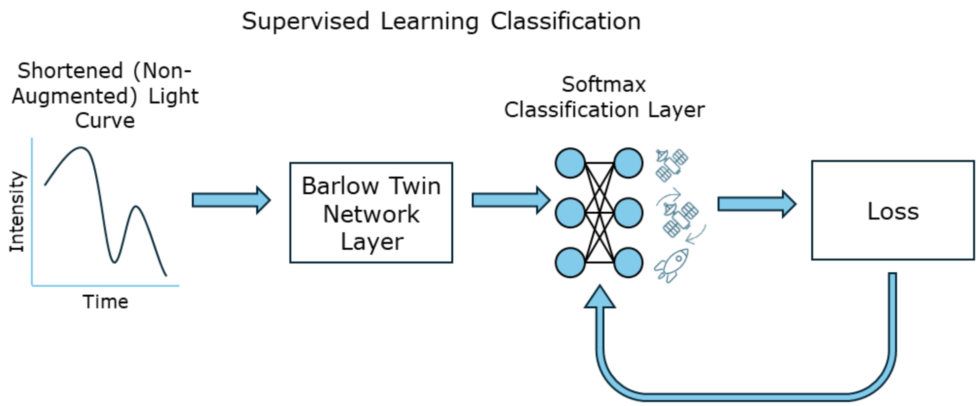
A summary for the implementation of Barlow Twins is as follows. Using the original data, which are light curves in this study, two distorted or augmented versions of each light curve are obtained, while ensuring the class of the object does not change. Both augmentations are passed through an encoder network to obtain representations. The encoded data are then passed through a projector layer, which consists of a series of dense layers. This results in two sets of embeddings from each augmented input. Then, the cross-correlation matrix between these embeddings is computed and the goal of the loss function is to make the cross-correlation matrix as close as possible to the identity matrix. A graphical summary of this architecture is in Fig. 3.2. The remainder of this section explains each of these steps in detail.

3.4.1 Dataset preprocessing

The first step in dataset preprocessing is unifying the length of the dataset. The light curves in this study have varying measurement durations. As a result, the length of the dataset has been unified for



(a) Barlow Twins summary



(b) Linear evaluation model overview

Figure 3.2: Overall network summary diagram.

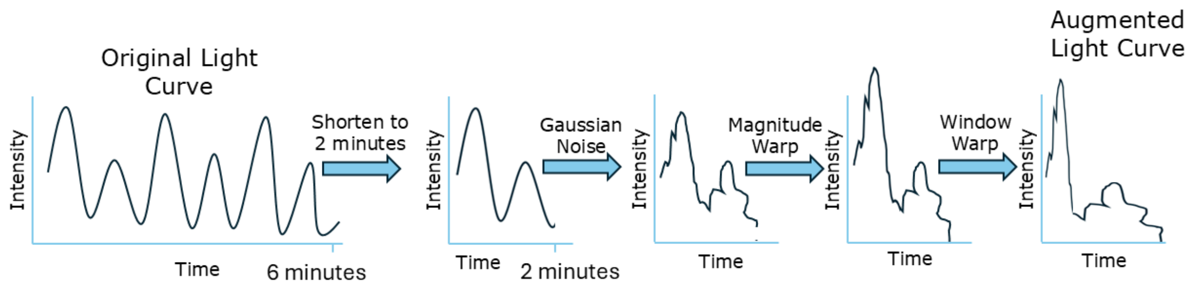


Figure 3.3: Overview of preprocessing and data augmentation.

ease of implementation into the ML algorithm. By evaluating the length of the available light curves, we determined that minimum durations of two, three, and five minute light curves, which represent 68%, 40%, and 10% of the dataset respectively, are appropriate measurement lengths to use for testing algorithm performance. As a result, three classifiers are trained, one for each of the selected durations. The minimum duration of two minutes was set as it minimizes the exclusion of shorter light curves given that most light curves are still included in the study. For light curves that are shorter than the specified length, they are removed from the input dataset. The maximum duration of five minutes was chosen to serve as a comparison point to the LSTM classifier presented in [1]. Since the light curves in the original dataset are sampled every 20 milliseconds, the light curves used in the classification task are down-sampled by a factor of five such that the data points are 100 milliseconds apart. A down-sampling factor of five was determined by finding the factor where the autocorrelation drops to $1/e$ or 36%. This factor was then tested to ensure that the performance of the classifier is not diminished.The down-sampling reduces the length of the dataset used in the neural network which speeds up training. There are 1200, 1800, and 3000 measured data points for each light curve in the two, three, and five minute light curve datasets. In this study, debris and unknown satellites are removed from the dataset as they will not be used for training the network. Debris represents 4% of the entire dataset, which is significantly smaller than the other classes. As for unknown satellites, they can be used in the classifier to determine whether they are stable or tumbling after the network has been trained. Then the data were split into training and test sets with an 80%-20% split. To increase the number of samples and to unify the number of light curves per class, light curves longer than the set duration with fewer light curves were split into multiple inputs. After dataset prepossessing, there is a total of 2515, 2538, and 699 light curves with measurement durations of two, three, and five minutes long, respectively.

3.4.2 Dataset Augmentation

One of the major aspects of the Barlow Twins network is its heavy reliance on augmentations for training. Given that the augmentations are used as inputs into the network, a set of two different but correlated light curves are required. Since the inputs are time series data and not images, the augmentation techniques applied in [90] are not compatible. The following augmentation techniques were used for the light curves: Gaussian noise for each light curve, as well as random window warping and random magnitude warping on part of the dataset 20% of the time. Gaussian noise was added to the light curves to simulate the effect of sensor noise or variations in measurement conditions. Gaussian noise was applied to make the trained Barlow Twins network robust against noisy data. Random window warping is randomly selecting a slice in the light curve, expanding, or narrowing it and then re-scaling the light curve back to its original length [99]. Random magnitude warping is altering the magnitude of the light curve at different points [99]. Since it is important to maintain the original classes of the light curves, augmentations that are too excessive are avoided. In this study, very minor augmentations were applied to the light curves. The random window and magnitude warping are not applied to all samples to add a variety in the augmentations to ensure robustness where some light curves have both magnitude and time warping, some have one approach applied and other only have Gaussian noise applied. A visual summary of the preprocessing and data augmentation steps on a sample light curve is in Fig. 3.3. Fig. 3.3 demonstrates both window and magnitude warping but not all light curves will have magnitude and time warping applied to them.

There are other augmentation approaches that can be tested in future studies. A simple approach is a moving window applied to the full length light curves to generate various subsamples originating from the same full-length measurement. Time embedding, where the time data is retained for training provides additional temporal context for the data which allows the network to better understand temporal trends in the data. Another approach is frequency domain augmentations, where filtering or phase shifts can be applied to introduce augmentation complexity. Lastly, mixup augmentation where two light curves are combined and interpolated to generate new samples can be applied to increase model robustness and reduce overfitting. This approach has been developed for image data in [100] but it can be implemented for time series data as well.

3.4.3 Barlow Twins network

The main functions of the Barlow Twins network are described in this section.

3.4.3.1 Encoder

The first part of the Barlow Twins architecture is the encoder network which computes representations of each of the augmented light curves. Similar to [90], we used a ResNet architecture. This network was originally developed for obtaining representations from images, and we have modified it to work for time series data. The architecture consists of four blocks of convolution with L2 regularization, batch normalization, Rectified Linear Unit (ReLu) activation, and skip connections. Then, global average pooling is applied.

3.4.3.2 Projector

Once each augmented light curve is encoded, the results are passed to the projector network which consists of two dense layers, each with 5000 neurons. Batch normalization and ReLu activation are also used in this function. When implementing this in code, a single function for building the Barlow Twins architecture was used which first calls the encoder network and then adds the dense layers. The embeddings of each augmentation resulting from the projector network are used as inputs to the loss function.

3.4.3.3 Loss function

The loss function for a Barlow Twins approach is what separates it from other architectures. As first described in [3], the loss function has two parts. The first part is called the invariance term. The goal of this part of the loss function is to make the diagonals of the cross correlation matrix between the embeddings of each augmentation equal to one which indicates that the augmented light curves are correlated or originating from the same original light curve. The second part of the loss function is the redundancy reduction term that aims to make the off-diagonals zero which means this part de-correlates parts of the embedding which reduces the redundancy between the embeddings of the augmentations. The loss function for Barlow Twins focuses on the redundancy reduction aspect between the augmentations only so it is a static approach. The redundancy reduction component is very important for building a robust network that is less prone to overfitting. A ResNet encoder was added to introduce a temporal element using skip conenctions, but for future studies, implementing an LSTM after the Barlow Twins architecture would be interesting to study given that LSTM is a sequential model typically used with time series data for capturing long-range dependencies.

3.4.3.4 Linear evaluation model

The Barlow Twins architecture only results in embeddings and does not perform classification. To reach the goal of light curve classification, a linear dense layer with softmax activation is added to the Barlow Twins model. The main aspect to note is that this step is only intended to perform classification and should not alter the features learned by the Barlow Twins model. As a result, the weights of the Barlow Twins model are set to be non-trainable such that they do not change during linear model training. The only layer that can be tuned at this stage is the final layer. Using a self-supervised algorithm that does not initially rely on labels allows for the network to heavily learn from the light curves directly as opposed to the labels. L2 regularization is applied to avoid overfitting. Unlike the Barlow Twins model, the linear evaluation model uses the original, unified length, light curves only and does not use the augmented version of the light curves. As a result, when testing a trained network, the original light curves are used with unified length but no other modifications.

For both the Barlow Twins and the linear model, Layer-wise Adaptive Moments optimizer for Batch (LAMB) training was used. Using LAMB resulted in faster convergence as opposed to Adaptive Moment (ADAM) estimation.

3.5 Results

This section shows the results from the LSTM and Barlow Twins LEO RSO classification models applied to two, three, and five minute long light curves. Table 3.2 shows the highest accuracy of both models obtained after hyperparameter tuning for the different light curve durations. The same network architecture as that in [1] has been used for this study but the network has been re-trained with a modified dataset to match the pre-processing performed on the Barlow Twins network. Using a duration

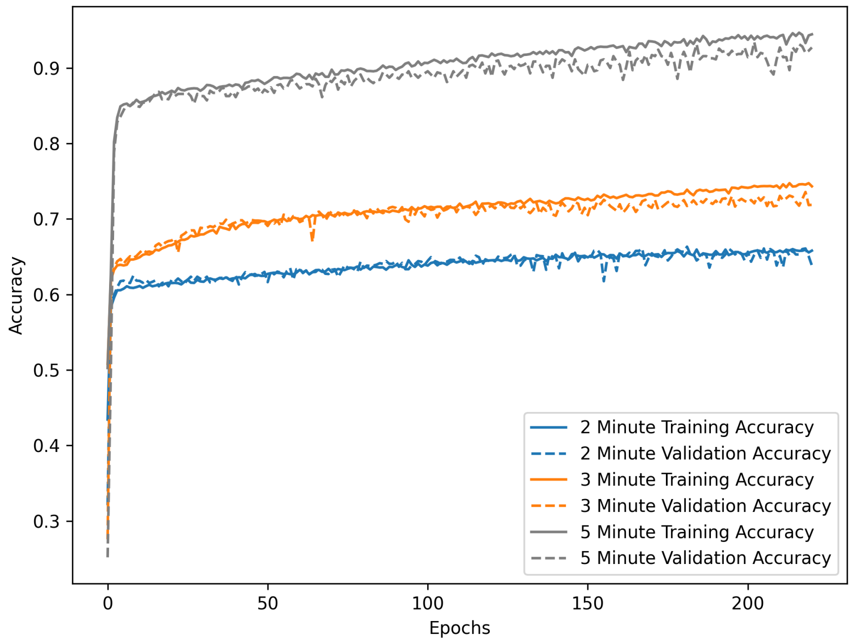
Table 3.2: Classification accuracy using LSTM and Barlow Twins for two, three, and five minute long light curves.

Duration (minutes)	LSTM	Barlow Twin
2	66%	75%
3	72%	85%
5	92%	97%

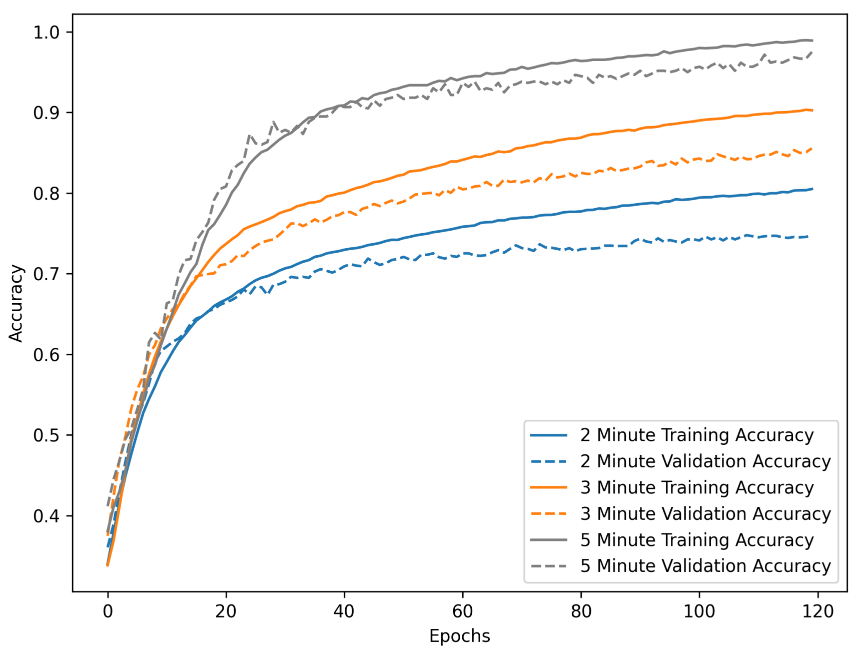
of five minutes gave the highest accuracy of 92% for the LSTM network and 97% for the Barlow Twins network. LSTM classifies a two minute light curve with 66% accuracy and a three minute light curve with 75% accuracy. On the other hand, Barlow Twins classifies a two minute light curve with 75% accuracy and a three minute light curve with 85% accuracy.

To evaluate the performance of the LSTM and Barlow Twins networks over the different epochs, the accuracy and loss plots are shown in Figs. 3.4 and 3.5 respectively. The plots of accuracy over epochs in Fig. 3.4 show the training and validation accuracy following each other with the Barlow Twins network having a gap that decreases as the length of the light curve increases. Similarly, for the Barlow Twins network the loss in Fig. 3.5 decreases as the light curve duration increases with the gap between the training and validation loss decreasing as well. Both accuracy and validation plots show the increase in performance as the number of epochs increase. However, for the LSTM loss, the gap between training and validation loss increases with the increase in epochs. It should be noted that the loss functions in the LSTM and Barlow Twins networks are not the same. As a result, we can compare trends between the training and validation loss values but cannot make assumptions about the difference in performance between the networks using this plot alone.

To assess how well the models perform in predicting each class, the classification precision, recall, and F1-scores per class are shown in Fig. 3.6. The LSTM network precision is between 60% and 98% while recall is between 50% and 92% and F1-score is between 57% and 92%. As for the Barlow Twins network, the precision is between 71% and 98%, the recall is between 70% and 100% while the F1-score is between 72% and 98%.

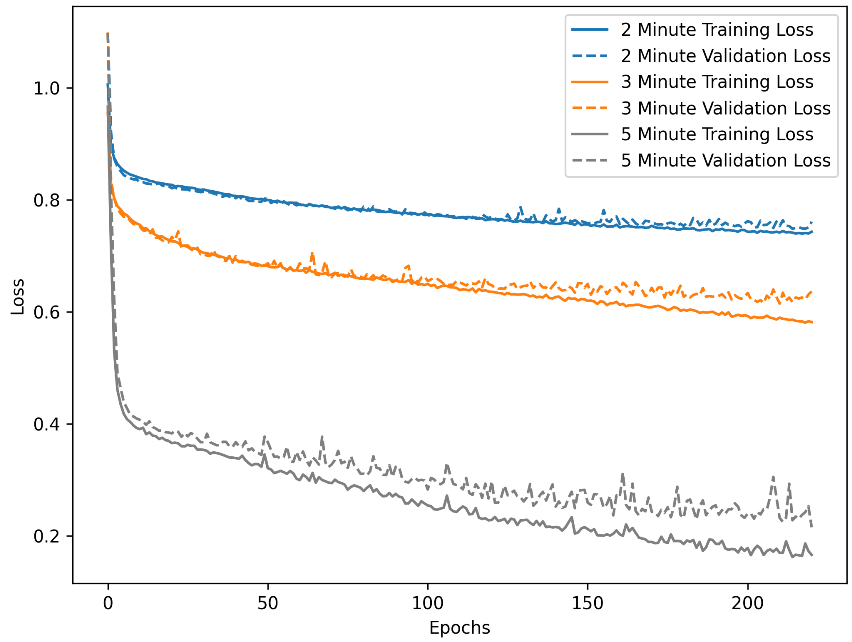


(a) LSTM

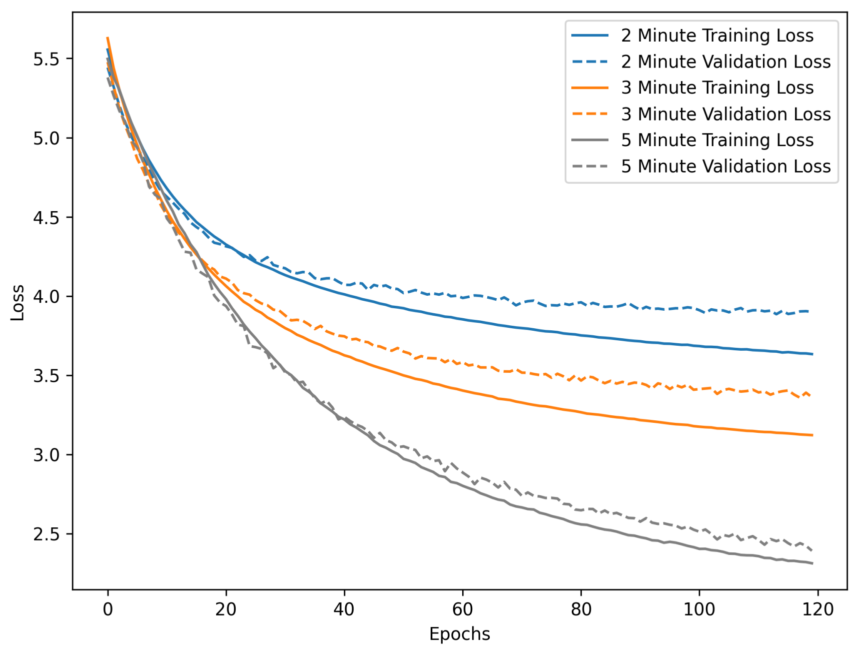


(b) Barlow Twins

Figure 3.4: Classification accuracy over epochs.

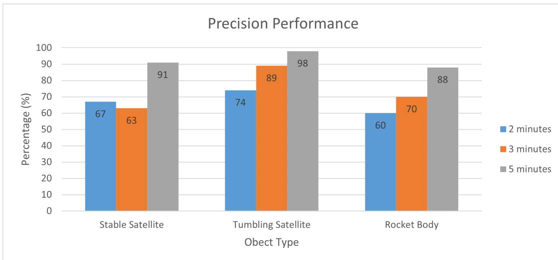


(a) LSTM

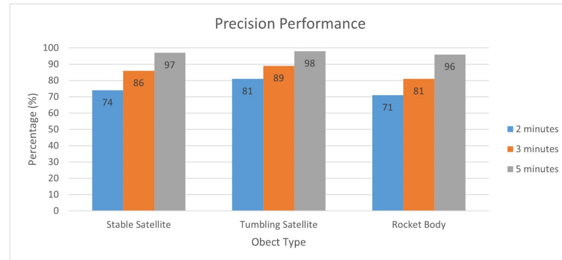


(b) Barlow Twins

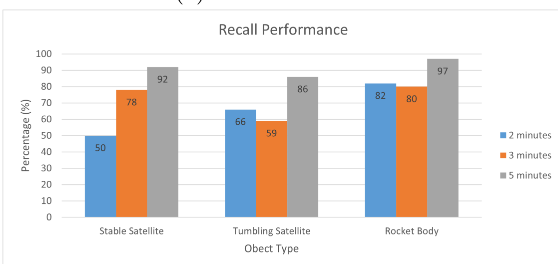
Figure 3.5: Classification loss over epochs.



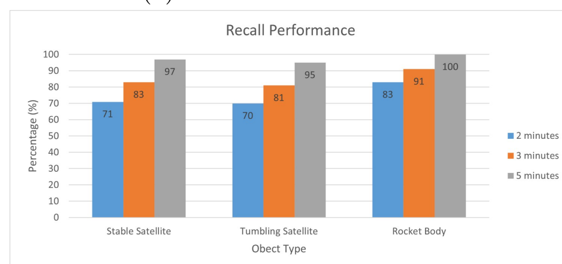
(a) LSTM Precision



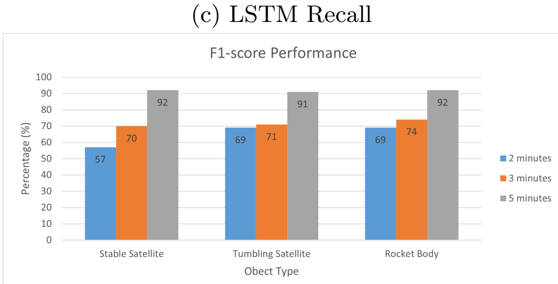
(b) Barlow Twins Precision



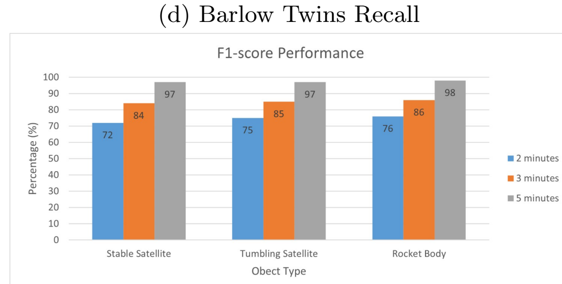
(c) LSTM Recall



(d) Barlow Twins Recall



(e) LSTM F1-score



(f) Barlow Twins F1-score

Figure 3.6: Precision, recall, and F1-score.

3.6 Discussion

Table 3.2 indicates a large improvement in the Barlow Twins architecture compared to the original LSTM architecture. The performance difference is most pronounced in the shorter durations of two and three minutes. When classifying two minute light curves, Barlow Twins has a 75% accuracy which is significantly more accurate than the LSTM with an accuracy of 66%. The accuracy increases with the increase in the length of the light curve. There are multiple reasons for this occurrence. First, it is more challenging for the network to determine the classes of shorter light curves correctly due to the insufficient measurements needed for a full rotation of an RSO to occur. If an RSO is captured for only half a revolution, then the light curves do not properly represent the brightness of all the RSO's faces which diminishes the performance of the classifier. Similarly, manually labelling light curves of RSOs with low rotation frequency is more difficult. Consequently, a higher percentage of the shorter light curves are potentially mislabeled. Additionally, the smaller size of the five minute dataset adds the possibility for the network to learn representations from the data more quickly and effectively and runs the risk of over-fitting much more quickly. To prevent over-fitting, L2 regularization is implemented in both the Barlow Twins network and the final classification layer. For all light curve durations, the Barlow Twins network outperforms the LSTM architecture. This is attributed to the redundancy reduction features of the Barlow Twins network where two augmentations are concurrently used. Another reason for the improved performance of Barlow Twins is due to the self-supervised approach to training the network with the supervised classification performed after the main network has been trained and the weights are frozen.

Analyzing the accuracy and loss plots over epochs in Figs. 3.4 and 3.5 indicate that the Barlow Twin network's tendency to overfit the data is reduced as the duration of the light curve is increased while the LSTM has a higher tendency to overfit the data with an increase in light curve duration. The LSTM's behavior is expected. As the length of the light curve increases, the size of the dataset decreases. As a result, the network becomes more prone to overfitting. The Barlow Twins behaves in a counter-intuitive manner where the larger dataset of shorter light curves has a larger generalization gap than the five minute light curves. This can be attributed to the two minute light curves being harder to discern and thus the network is more likely to memorize the inputs as opposed to learning from them efficiently. For LSTM, the impact of the size of the dataset is more evident whereas for the Barlow Twins, the main impact factor is the ease of extracting features which depends on the length of each light curve.

Fig. 3.6 indicates that the performance across the different classes is generally consistent with performance on rocket bodies having the highest F1-score across both networks in the three different durations. Regarding the LSTM network, the F1-score increases with the increase in duration but this is not true for precision and recall. For stable satellite classification, the two minute light curves result in a slightly higher precision and much lower recall than the three minute light curves. This indicates that the network is mis-classifying many stable satellites when using two minute light curves. Using three minute light curves might allow the LSTM to better learn the characteristics of stable satellites.

As for the Barlow Twins network, there is an overall balance between the precision and recall values across all classes with stable and tumbling satellites having a minor increase in precision over recall. Rocket bodies have the opposite trend where the recall is higher than the precision. Despite the increased performance of rocket body classification, there is slightly less precision than the stable and tumbling satellites. This indicates that the network mis-classifies some satellites as rocket bodies more often than mis-classifying rocket bodies as satellites. Most satellite light curves tend to have unique trends that are missing from rocket bodies so it is less likely for a rocket body to be labelled as a satellite. However, satellite light curves, both stable and tumbling have more variation in their patterns so some of the satellites could be mistaken as rocket bodies. When comparing the precision and recall values of the different light curve durations, the difference becomes much less noticeable for the five minute light curves. This is attributed to the same reasons for improved network performance where the distinction across classes is much more pronounced for the five minute light curves. Furthermore, there is roughly a 10% increase in precision, recall, and F1-score between the two and three minute light curves and between the three and five minute light curves. The longer the light curve, the more information is captured and the easier it is to classify. Further improvements to the Barlow Twins architecture will be made to improve the accuracy of the light curve classifier when using shorter light curves.

3.7 Conclusion

In conclusion, a novel application of the Barlow Twins network has been demonstrated for short LEO RSO light curve classification. This approach works with light curves as short as two minutes with an accuracy of 75% while the technique demonstrates exceptional performance with the five minute light curves at an accuracy of 97%. Although the algorithm is trained on augmentations, once the network is trained, the test light curves are used directly without any augmentation. With an accuracy of 75% for two minute inputs, improvements can be made for the algorithm to yield better performance using short duration light curves.

Future work includes evaluating more complex augmentation approaches to extract light curves that have more pronounced differences while ensuring the classification label remains consistent. Other work includes adding a fourth classification category for debris. Additionally, the classification algorithm can be used in the future to determine whether the RSOs labelled as "Unknown Satellites" are stable or tumbling satellites. Lastly, other networks will be tested within the Barlow Twins instead of a ResNet, such as an LSTM, transformer, and VAE. These approaches might allow for improved accuracy with short light curves.

Chapter 4

Understanding Light Curve Classification Using eXplainable Artificial Intelli

4.1 Introduction

In the previous two chapters, we investigated the application of SVM, LSTM, and Barlow Twins to classify LEO RSO light curves. Each light curve is classified as a stable satellite, tumbling satellite, or rocket body. The best performance results come from the Barlow Twins architecture with a ResNet encoder, yielding accuracy values of 75%, 85%, and 97% for two, three, and five minute light curves, respectively. While these results are promising, the "black box" nature of this model poses challenges in understanding the underlying factors driving these classifications. Even though the network's performance has been quantitatively assessed using metrics such as accuracy, precision, recall, and F1-score, there is a lack of a qualitative assessment linking these results back to the original light curves. This includes understanding why certain cases succeed while others fail. This lack of transparency impairs our ability to reliably interpret the network's decisions [101], which is crucial for the proposed light curve classifier to be effectively used by spacecraft operators. To address this, we integrated XAI techniques into our framework. XAI provides tools and methodologies to visualize and quantify the ML model performance, highlighting the features and attributes influencing classification outcomes. By leveraging XAI, we aim to identify the specific characteristics in light curves that the network associates with stable satellites, tumbling satellites, or rocket bodies. Another potential approach to increasing our understanding of the network is to perform a manual or analytical review of input-output relationships. In the case of light curve classification, this includes observing the successful and failed classification cases to see if a light curve needs certain features or trends for it to be successfully classified. While manual assessment has the potential to provide useful insights, XAI methodologies will enable a deeper, data-driven understanding of the network's decision-making process.

4.2 Methodology

Many algorithms exist in the field of XAI. While most of these algorithms have been developed for networks trained on images, they can also be applied to 1D time series classification. A survey of XAI techniques for time series classification is provided in [102]. As per [102], there are many types of XAI approaches and they can be categorized in different ways. XAI models can provide local and global explanations. Local explanations clarify the network's performance for specific inputs, while global explanations provide an overall understanding of the network's behavior. Another type of category is post-hoc methods, which are model-independent, and ante-hoc methods, which are directly interpretable models. XAI techniques can be model-specific, functioning only with certain models, or model-agnostic, applicable to any classification network. For time series classification, the study in [102] categorizes XAI approaches into three types. The first, time points-based methods, provide explanations for each point in the data. The second, sub-sequence-based techniques, highlight unique features or characteristics in parts of the time series data. Finally, instance-based methods analyze the full time series input, or in this case, the entire light curve. Before selecting an XAI technique, we must first define the analysis or results we expect to obtain. In this study, we aim to generate two key outputs to better understand how the Barlow Twins network evaluates light curves, namely saliency maps and prototypes. Saliency maps, derived from time point-based XAI approaches, overlay the input light curve with the weight or importance of each element. These maps highlight which parts of the light curve the model focuses on when making a decision, resulting in a plot with a color map that visualizes the features influencing the classification. This helps clarify the model's decision-making process. The second analysis focuses on prototypes, which are produced using instance-based approaches. Prototype-based XAI analysis determines one sample per classification category (i.e., stable satellite, tumbling satellite, and rocket body) that best represents the class. These prototypes provide a clear reference point for understanding the criteria used by the model to classify different RSOs.

To obtain saliency maps and prototypes, several XAI approaches can be applied. Some methods used to generate saliency maps include Gradient-weighted Class Activation Mapping (Grad-CAM), Local Interpretable Model-agnostic Explanations (LIME), and SHapley Additive exPlanations (SHAP). Grad-CAM was introduced by [103] for visualizing CNN image heatmaps by utilizing the gradients from the last convolutional layer. It is a local, post-hoc, model-specific method that generates heatmaps for each classification category. In this study, Grad-CAM will be used to generate saliency maps for the Barlow Twins network, which is based on a ResNet architecture. A key advantage of Grad-CAM is that it retains temporal information when applied to time series data, such as light curves, preserving the sequence of brightness values. However, it is limited to CNN-based models. SHAP, first introduced in [104], uses game theory to assign each feature in the network an importance score for each light curve prediction. Although powerful, SHAP is computationally expensive, particularly for deep networks [105]. As for LIME, it uses sparse linear models to approximate the network being studied [106]. Using a sparse model where only a portion of the features are used at a time allows for model simplicity. By introducing perturbations in the data, LIME can observe how these perturbations impact the linear model. While simple, LIME struggles with complex models like deep networks, especially in time series data where the relationship between different time steps is not properly considered in the local structure created by LIME. Both SHAP and LIME are model-agnostic, local, and post-hoc methods, but Grad-CAM was chosen for this study due to its ability to leverage network gradients and produce detailed localization maps, making it a suitable method for complex networks. As a result, it does not encounter the same limitations faced by SHAP and LIME. The saliency map from Grad-CAM highlights the specific time steps that are most influential in the model's decision-making process. By interpreting these plots, we can identify which aspects of the light curve are critical for distinguishing between stable satellites, tumbling satellites, and rocket bodies.

For prototype generation, K-means clustering will be implemented. This involves applying K-means to obtain three clusters, with the most representative light curve from each cluster identified as the element closest to the cluster center. K-means uses distance similarity to arrange clusters. While there are many XAI approaches for prototype generation such as Dual Prototypical Shapelet Networks (DPSN) and Prototype Sequence Network (ProSeNet), K-means is chosen as a simple and efficient approach for grouping the light curves using the scores from the trained network directly. K-means is a model agnostic, post-hoc approach that provides a global instance-based explanation of the network.

4.3 Results

The results shown in this section are for two minute light curves. This duration is chosen to better assess more challenging classification situations as longer duration light curves have higher classification accuracy. Given that brightness is an inverted scale, where lower values indicate brighter objects, all light curves shown in this section are plotted with an inverted y-axis such that a peak shows an increase in brightness and a trough indicates a brightness decrease. For the two minute light curve classification, 595 stable satellites, 584 tumbling satellites, and 697 rocket bodies were correctly classified with those categories. The stable and tumbling satellite classes each have 838 light curves and the rocket body class has 839 light curves.

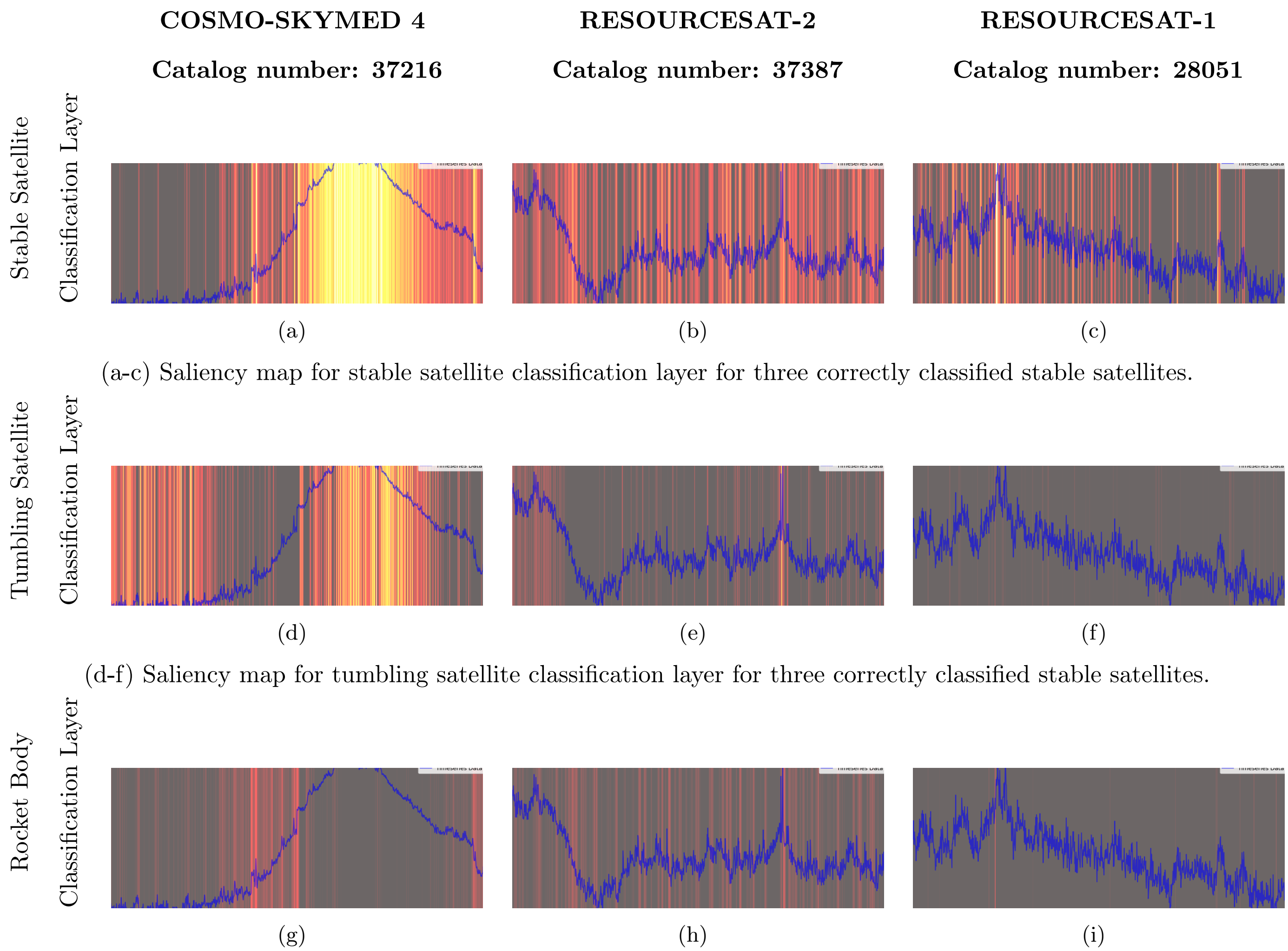
4.3.1 Grad-CAM

Grad-CAM is used to assess the network's performance for both correctly classified and misclassified light curves. By generating saliency plots of correct classifications, we can identify the features the network is looking for to make accurate predictions. By studying misclassified instances, we gain a deeper understanding of the reason for the misclassification.

4.3.1.1 Correctly Classified Light Curves

The Grad-CAM saliency plots will be grouped by the class of the input light curve. Figure 4.1 shows the saliency plots for three example stable satellites for the stable satellite, tumbling satellite, and rocket body classification layers.

The saliency map contains a color bar with an importance score at each measurement for each of the three different classification layers. The resulting classification in the network represents the object type with the highest importance scores throughout the light curve. The colors indicate the following: grey values represent negligible importance, red indicates moderate importance and yellow indicates very high importance scores. In the first example satellite, COSMO-SKYMED 4, the singular peak has



(a-c) Saliency map for stable satellite classification layer for three correctly classified stable satellites.

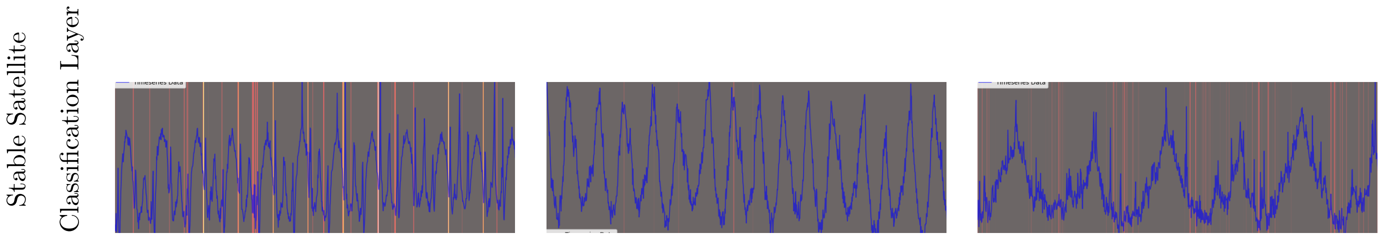
(d-f) Saliency map for tumbling satellite classification layer for three correctly classified stable satellites.

(g-i) Saliency map for rocket body classification layer for three correctly classified stable satellites.

Figure 4.1: Grad-CAM Saliency Map for three sample stable satellites.

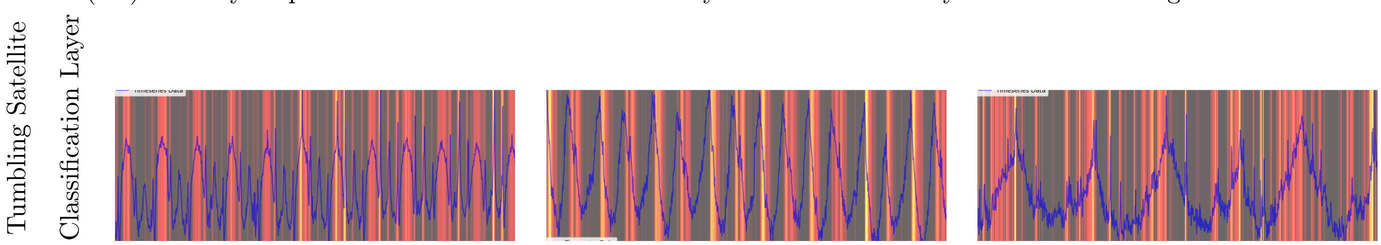
a high importance score in the stable satellite classification with the tumbling satellite layer showing some importance at the same peak but not at the same strength. On the contrary, the rocket body layer shows relatively low scores across the light curve. For RESOURCESAT-1 and RESOURCESAT-2, there is moderate importance across the light curves with some points of high importance when looking at the stable satellite layer. For the tumbling satellite and rocket body layers, there is mostly negligible importance with some small areas of moderate importance. This indicates that all three light curves will be classified, correctly, as stable satellites. For COSMO-SKYMED 4, the reason for that classification is the large increase in brightness at one point which is most likely due to reflection from the satellite's solar array. For RESOURCESAT-1 and RESOURCESAT-2, the presence of some peaks or troughs that do not repeat result in the correct classification. The next type of light curve class to be studied is a tumbling satellite. Figure 4.2 shows the Grad-CAM saliency map for three tumbling satellites.

TOPEX (POSEIDON)	**MIDAS 6**	**OPS 5721**
Catalog number: 22076	**Catalog number: 00574**	**Catalog number: 09415**



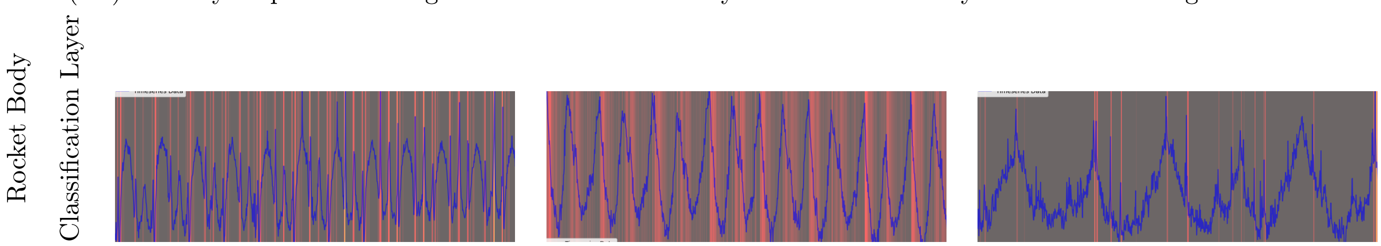
(a) (b) (c)

(a-c) Saliency map for stable satellite classification layer for three correctly classified tumbling satellites.



(d) (e) (f)

(d-f) Saliency map for tumbling satellite classification layer for three correctly classified tumbling satellites.

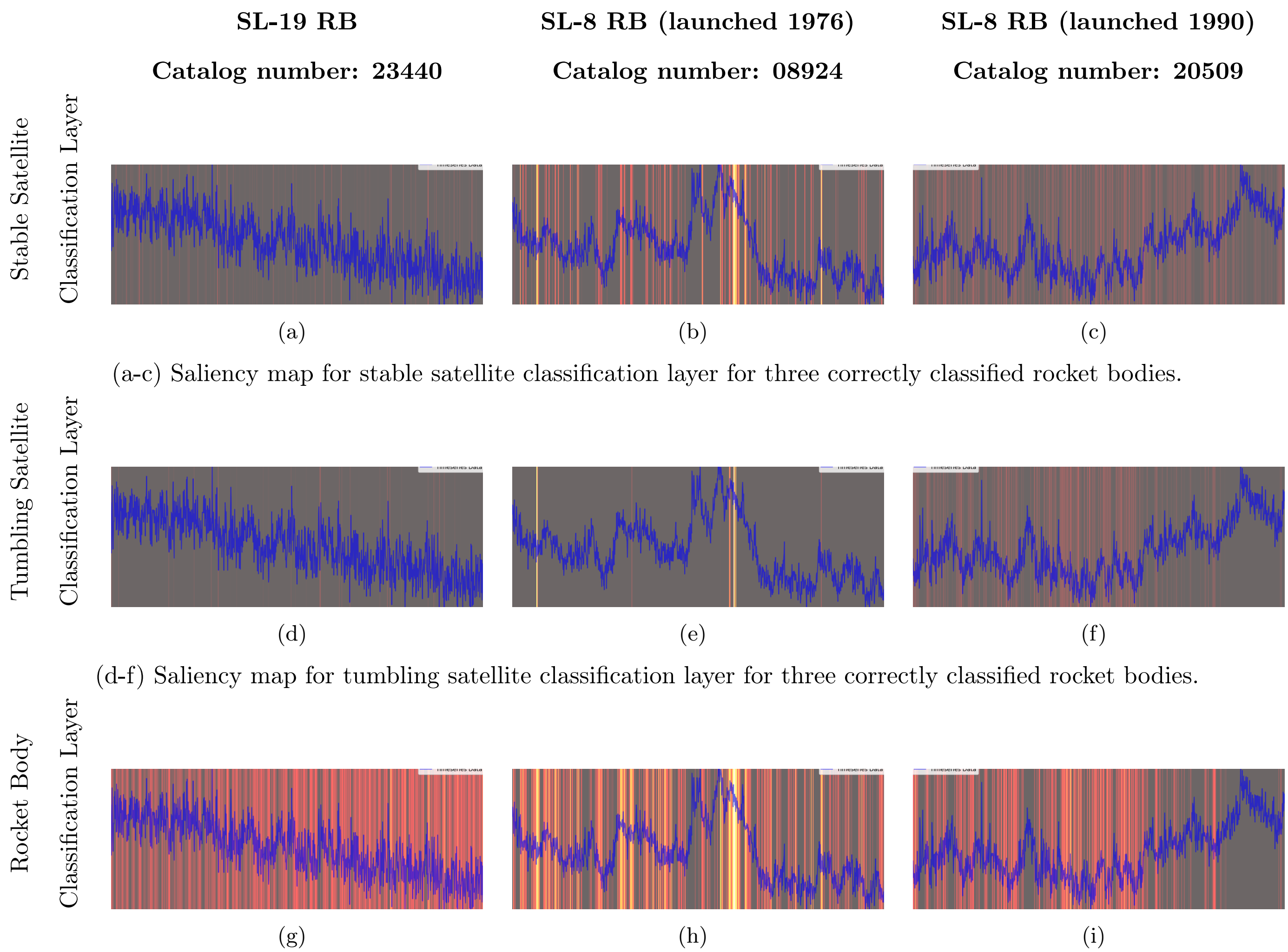


(g) (h) (i)

(g-i) Saliency map for rocket body classification layer for three correctly classified tumbling satellites.

Figure 4.2: Grad-CAM Saliency Map for three sample tumbling satellites.

Figure 4.2 shows moderate to high importance scores for the tumbling satellite class at the peaks and troughs. The stable satellite and rocket body classes show moderate to weak scores throughout most of the light curve. This classification clearly shows the distinctive features of the tumbling satellite class. Multiple consecutive peaks with similar magnitude in a short period of time are a strong indicator of tumbling satellites. Lastly, rocket body saliency maps are studied. Figure 4.3 show the Grad-CAM saliency maps for three sample rocket bodies.



(a-c) Saliency map for stable satellite classification layer for three correctly classified rocket bodies.

(d-f) Saliency map for tumbling satellite classification layer for three correctly classified rocket bodies.

(g-i) Saliency map for rocket body classification layer for three correctly classified rocket bodies.

Figure 4.3: Grad-CAM Saliency Map for three sample rocket bodies.

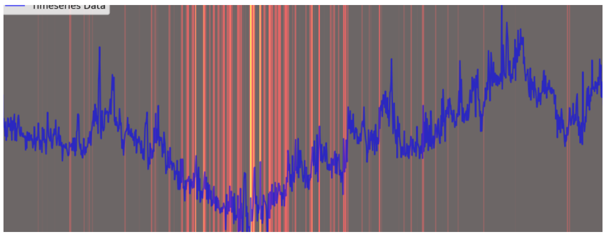
The results for the rocket body classification are very different from the other two classes. While the stable satellite and tumbling satellite saliency maps showed very high importance scores at certain portions of the light curve, the rocket body saliency map almost always has a medium importance for the true rocket body classification. Both stable satellite and tumbling satellite classification have an almost zero importance score throughout most of the light curve with some areas with moderate importance. While there are not as many regions with a very high importance score for the rocket body layer, the score is relatively consistent and is much higher than the values for the stable and tumbling satellite classification layers. This might indicate that the network considers a rocket body as representative of a “non-satellite” category such that it is neither a stable satellite nor a tumbling satellite instead of relying on one particular feature.

4.3.1.2 Misclassified Light Curves

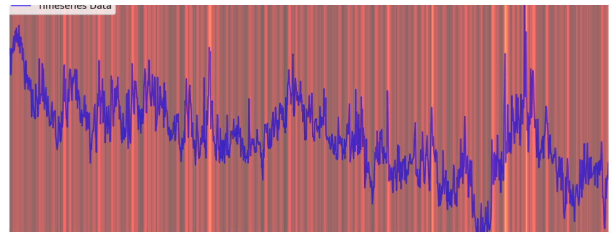
With the understanding gained through the assessment of correctly classified light curves, this section will explore the misclassified light curves to observe instances where the network learns incorrect features or the physical features of the RSO result in light curves that are not taken into consideration within the network. The first set of light curves that will be assessed are misclassified stable satellites. Figure 4.4 shows two stable satellite light curves where one light curve is misclassified as a tumbling satellite and the other is misclassified as a rocket body.

Figure 4.4 shows instances of misclassified stable satellites resulting due to unexpected trends or features that the network has not learned. For the object classified as a tumbling satellite, there are different types of peaks resulting from multiple different features or parts of the satellite. There seem to be three main peaks most likely associated with reflection off KAZEOSAT 1’s three solar panels. Regarding the light curve of LANDSAT 8 that is misclassified as a rocket body, the issue seems to occur due to the lack of prominent peaks except for a very narrow peak with a significant brightness increase around the end of the light curve that the network does not appropriately take into account. A small number of prominent features is indicative of a rocket body. This results in the network having moderate importance scores in the stable satellite layer with moderate to high scores across the rocket body layer. The scores in the stable satellite layer are significant enough that the RSO would have been classified correctly had the importance score of the rocket body layer been lower. The next sequence of saliency plots that will be studied are those resulting from misclassified tumbling satellites as shown in Figure 4.5.

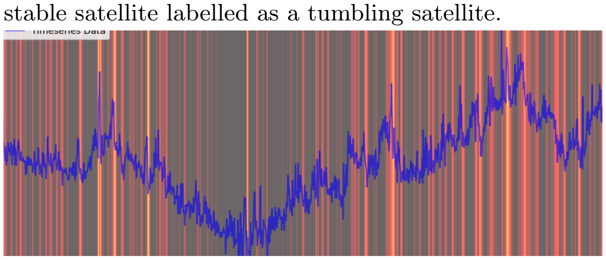
By visually assessing the light curve of TIROS N in Figure 4.5, the reason for misclassification becomes apparent. The light curve does not show multiple peaks with similar magnitude in quick succession which is typical of tumbling satellites. As for the light curve of GENESIS 2, it does indeed look like a light curve belonging to the rocket body. The almost linear decrease in brightness with few distinct peaks are representative of rocket body light curves. This situation indicates a configuration that is not represented in most tumbling satellite light curves. This occurs because GENESIS 2 is cylindrical in



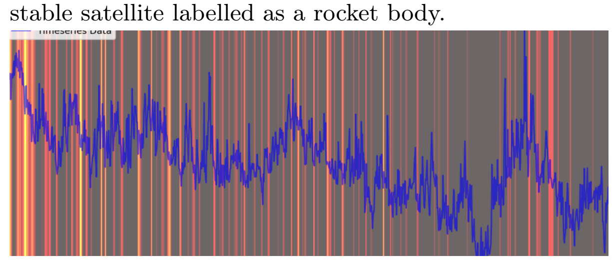
(a) Stable satellite classification layer saliency map for a stable satellite labelled as a tumbling satellite.



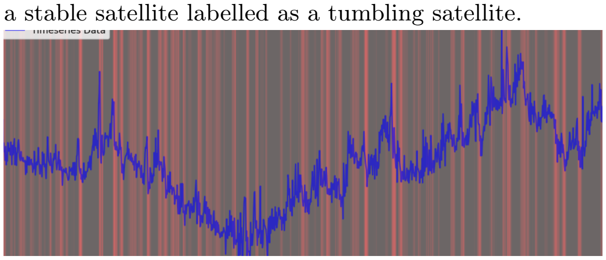
(b) Stable satellite classification layer saliency map for a stable satellite labelled as a rocket body.



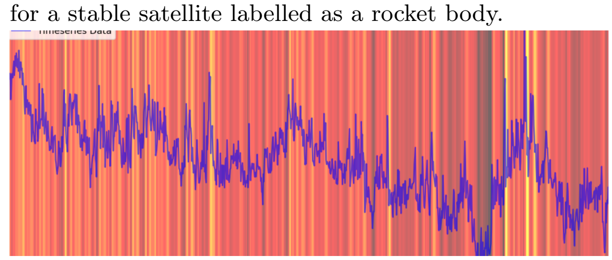
(c) Tumbling satellite classification layer saliency map for a stable satellite labelled as a tumbling satellite.



(d) Tumbling satellite classification layer saliency map for a stable satellite labelled as a rocket body.



(e) Rocket body classification layer saliency map for a stable satellite labelled as a tumbling satellite.



(f) Rocket body classification layer saliency map for a stable satellite labelled as a rocket body.

Figure 4.4: Grad-CAM Saliency Map for two misclassified stable satellites.

shape similar to a rocket body. The last type of misclassified light curves represents those of rocket bodies that have been classified as either stable or tumbling satellites. Figure 4.6 shows saliency plots for misclassified rocket body light curves.

Figure 4.6 shows two examples of misclassified rocket body light curves. Both light curves are atypical for rocket bodies. For SL-3 RB that has been misclassified as a stable satellite, the light curve has one large and wide trough at the beginning followed by a smaller and narrower peak at the end. This pattern is common for stable satellites where there are peaks or glints but there is no repetition. There is no consistent increase or decrease with brightness that is typically associated with rocket body light

TIROS N

Catalog number: 11060

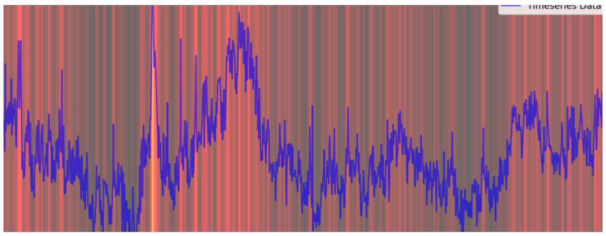
Predicted Class: Stable Satellite

GENESIS 2

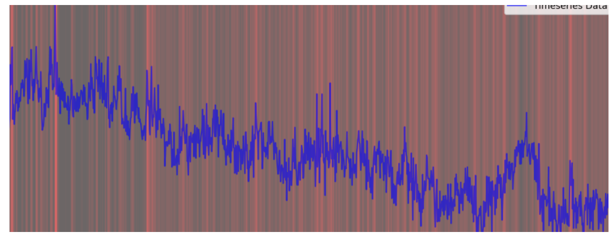
Catalog number: 31789

Predicted Class: Rocket Body

Stable Satellite Classification Layer

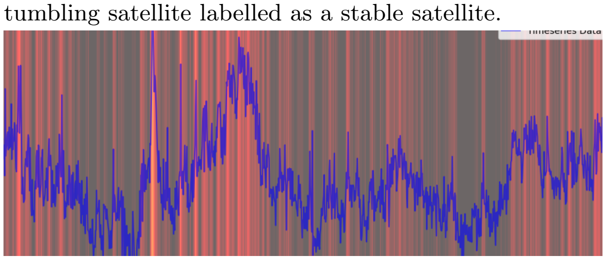


(a) Stable satellite classification layer saliency map for a tumbling satellite labelled as a stable satellite.

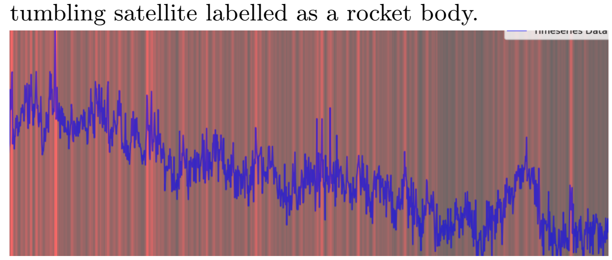


(b) Stable satellite classification layer saliency map for a tumbling satellite labelled as a rocket body.

Tumbling Satellite Classification Layer

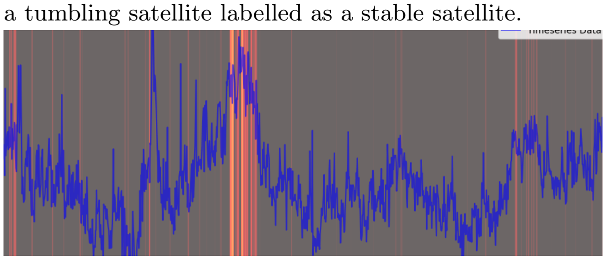


(c) Tumbling satellite classification layer saliency map for a tumbling satellite labelled as a stable satellite.

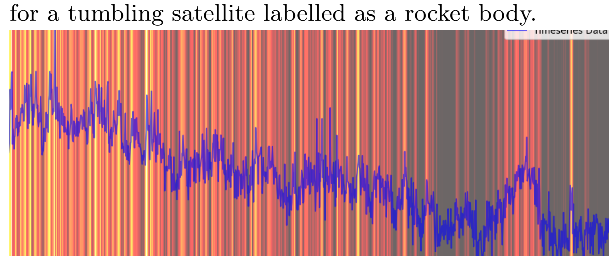


(d) Tumbling satellite classification layer saliency map for a tumbling satellite labelled as a rocket body.

Rocket Body Classification Layer



(e) Rocket body classification layer saliency map for a tumbling satellite labelled as a stable satellite.



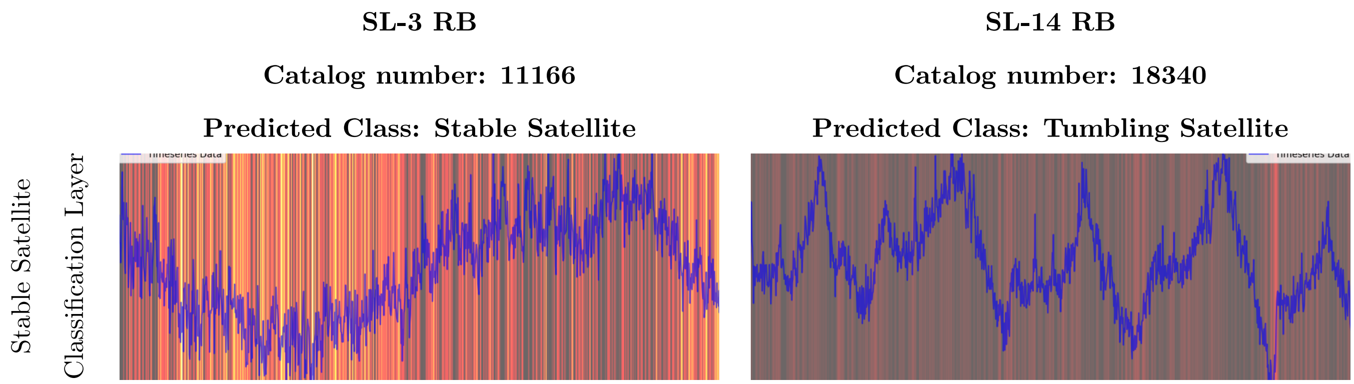
(f) Rocket body classification layer saliency map for a tumbling satellite labelled as a rocket body.

Figure 4.5: Grad-CAM Saliency Map for two misclassified tumbling satellites.

curves. A similar situation occurs for SL-14 RB that has been misclassified as a tumbling satellite. The saliency plot shows moderate to high importance scores in the tumbling satellite layer near the troughs. The light curve has multiple repetitive peaks which are typically associated with tumbling satellite light curves.

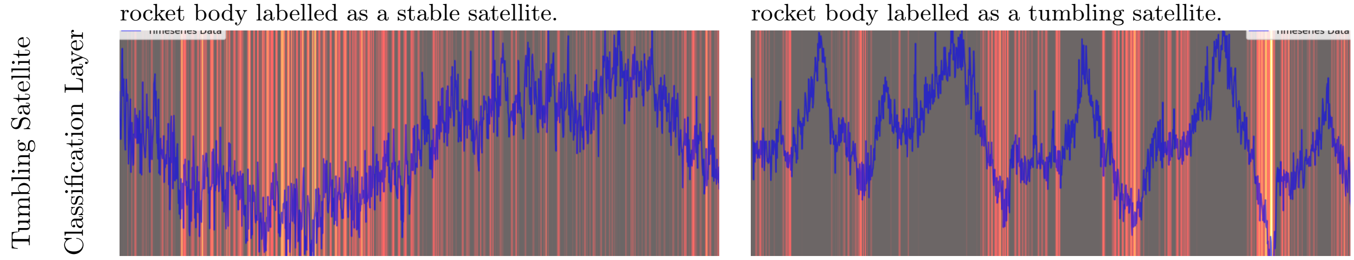
4.3.2 K-means

To compliment our understanding of the Barlow Twins network, prototypes generated from K-means are generated. These are the most representative samples of each class. Figure 4.7 shows prototypes of



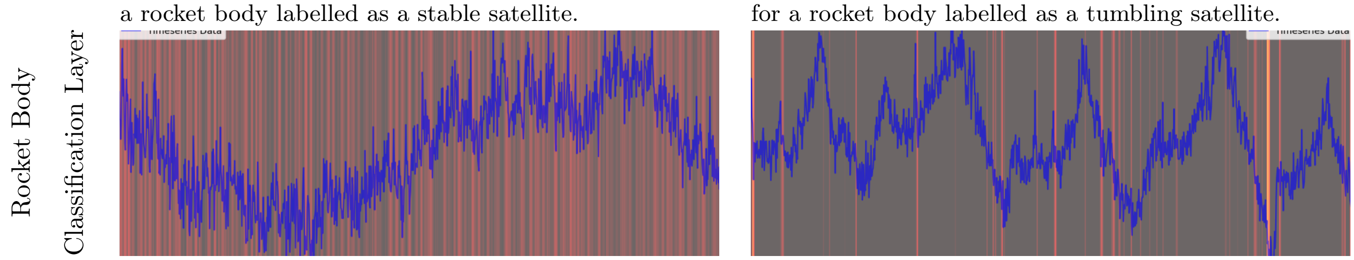
(a) Stable satellite classification layer saliency map for a rocket body labelled as a stable satellite.

(b) Stable satellite classification layer saliency map for a rocket body labelled as a tumbling satellite.



(c) Tumbling satellite classification layer saliency map for a rocket body labelled as a stable satellite.

(d) Tumbling satellite classification layer saliency map for a rocket body labelled as a tumbling satellite.



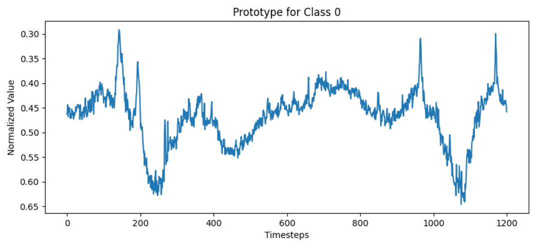
(e) Rocket body classification layer saliency map for a rocket body labelled as a stable satellite.

(f) Rocket body classification layer saliency map for a rocket body labelled as a tumbling satellite.

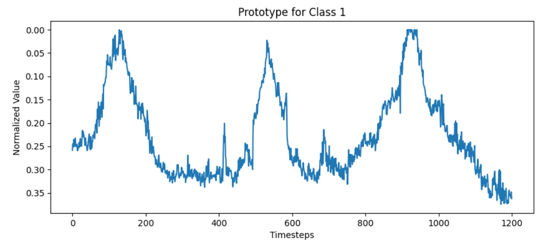
Figure 4.6: Grad-CAM Saliency Map for two misclassified rocket bodies.

a stable satellite, tumbling satellite, and rocket body as determined by K-means clustering.

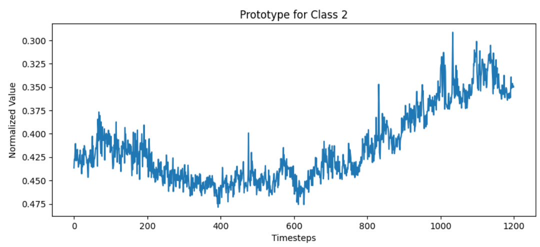
Figure 4.7 shows representative samples of a stable satellite, tumbling satellite, and a rocket body. The results match the understanding gained from the saliency map study. Stable satellites have few sparse peaks and troughs. The peaks are mostly due to reflections from different spacecraft components such as solar panels and antennas. Tumbling satellites have many peaks in succession, which occur due to the quick rotation of the spacecraft. Rocket bodies have no definitive pattern but instead, typically show an almost linear increase or decrease in brightness. These results show examples of the light curves that best represent each category. The results are insensitive to time direction but are sensitive



(a) Stable Satellite



(b) Tumbling Satellite



(c) Rocket Body

Figure 4.7: Light curve prototypes as determined by K-means clustering.

to cropping and rotation rate as the magnitude of the largest peaks would change.

4.4 Future Improvements to Barlow Twins

The information provided by XAI highlights which part of the light curve is most significant for classification. Through the use of Grad-CAM and K-means, a better understanding of the defining features of each of the three light curve classes was provided, which will provide the user with more knowledge to allow for label verification. By studying the misclassified light curve instances using Grad-CAM, we get a better understanding of the gaps in the network and the collected data. XAI is used to increase the confidence and trustworthiness of the Barlow Twins classifier. By combining quantative metrics such as accuracy with a better network understanding using XAI, the reliability of the network improves as we can combine the classification results with specific features on the light curve that influenced the classifier.

There are other ways to implement XAI beyond what was described in this chapter. First, XAI can be used to improve augmentation strategies. By using Grad-CAM to assess the importance scores of different parts of the light curve, the user can make an informed decision on which augmentations are useful and which negatively impact the Barlow Twins performance. Additionally, XAI can allow for improvements to the network architecture. By testing different architectures and examining the respective Grad-CAM plots, a better understanding of the strengths and weaknesses of the different approaches can be utilized to build a robust network.

4.5 Conclusion

In conclusion, this study demonstrates the application of XAI techniques to the Barlow Twins light curve classification for two minute light curves. A saliency plot is generated with Grad-CAM that provided a clear understanding of what features each classification category is looking for to assign a high classification score in that category. Grad-CAM proved to be instrumental in revealing insights from both successful and failed classification cases. For successful cases, it highlighted the critical features of light curves essential for accurate classification. In contrast, by examining failed instances, we gain a deeper understanding of scenarios where the light curves of one class could appear similar to those of another class (such as the SL-14 RB light curve appearing similar to a tumbling satellite light curve). These findings also revealed the network's limitations, guiding where improvement can be made. Further, the results from Grad-CAM are solidified with prototypes generated from K-means clustering that show representative samples from each class.

Grad-CAM and K-means offer distinct methods for explaining the performance of ML models, each providing valuable, yet different insights. Grad-CAM results in saliency maps for each classification layer per input light curve and K-means results in a single sample prototype per class that is the most representative light curve from each of the three classes. While it is not possible to perform a direct comparison between the two approaches, both techniques offer unique insights into the performance of the network and the light curve characteristics that are most useful for classification. The results from both approaches compliment each other where the features from each prototype determined by K-means follows the same trends as the features highlighted through the Grad-CAM saliency maps. In summary, the light curves of stable satellites exhibit a few sparse peaks with minimal repetition indicative of stable behavior, where only a few rotations occur per light curve. Tumbling satellites, on the other hand, show multiple consecutive peaks. Rocket bodies tend to display light curves without noticeably prominent peaks, instead showing a gradual increase or decrease in brightness, often in an almost linear pattern.

By enhancing interpretability through XAI, we aim to not only validate the Barlow Twin model's quantitative performance metrics, but also establish a qualitative link back to the original light curves. Gaining a full understanding of what the light curve classifier is looking for in the input will enable spacecraft operators to extract actionable insights. This approach will facilitate informed decision-making and improve operational reliability in classifying LEO RSOs.

PART II: SSA Camera Sensitivity & Simulation Study

For effective SSA that requires frequent monitoring of specific RSOs, space-based imaging is essential. While high spatial resolution cameras are commonly used for performing SSA studies, there is a growing interest in using low spatial resolution imagers due to the lower cost and commercial availability of such cameras. Additionally, a large number of these cameras already exist in orbit, such as star trackers. Although typically used for attitude determination, these cameras can also be employed for RSO observations and light curve extraction.

In this part, we describe two key areas where new contributions have been made to enable SSA missions using low spatialresolution cameras. The first contribution is a detectability analysis of different cameras for SSA, that was conducted during mission planning. The second contribution focuses on mission operations, specifically the use of simulated images for prediction. In our discussion of prediction algorithms, we examined a unique observation mode: anti-sun pointing.

Chapter 5 is based on the research collaboration with Magellan Aerospace Corporation to support the ongoing Redwing Mission. Redwing is Canada's next generation SSA microsatellite mission that will be launched in 2026 as a follow-up to Sapphire and NEOSSat. Redwing will perform RSO tracking and near real-time tasking [107]. Results presented herewith are based on a related trade study for a hypothetical photometric SSA mission. The chapter is based on the following publication: R. Qashoa *et al.*, "Wide Field of View (FOV) Imagers for Co-Orbiting Object Detection," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2024. [Online]. Available: `https://amostech.com/TechnicalPapers/2024/Space-Based-Assets/Qashoa.pdf`. The chapter focuses on the FAI, PCO, and IDS cameras. The PCO was flown on the Resident Space Object

Near-Space Astrometric Research (RSONAR), and both PCO and IDS were flown on the RSONAR II stratospheric balloon missions (more details on the mission are in [11]). The analysis in Chapter 5 is one of the earliest steps performed in planning a mission as it allows us to assess the capability of different optical cameras in observing RSOs.

Chapter 6 is based on a collaboration with Magellan Aerospace Corporation, C-CORE, and their partners under the Special Processes and Advanced Computing Environment for Detection, Unambiguity, Surveying, and Tracking (SPACEDUST), and is based on the paper R. Qashoa *et al.*, "RSO Simulations with Anti-Sun Pointing Predictions," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2023. [Online]. Available: `https://amostech.com/TechnicalPapers/2023/Poster/Qashoa.pdf`. This chapter introduces the concept of applying anti-sun pointing in an RSO optical image simulator to be able to perform predictions of what FAI will be observing, given that it is currently oriented anti-sun. Such configuration is favorable for photometric analysis, so this simulation environment can be used for photometric mission simulations where the observing spacecraft is oriented anti-sun. Simulations allow us to better understand what objects will be in the FOV of the observer, which in turn allows us to focus on specified targets that are expected to yield the best light curves, or those that are expected to be tumbling the most.

Analysis presented in Chapter 7 is also from work on SPACEDUST. The work presented in this chapter is currently under review as R. Qashoa *et al.*, "Demonstration of RSO Simulation, Prediction and Detection Using a Low-Resolution Imager in Anti-Sun Pointing," *Advances in Space Research*, Under Review 2024. This work is a follow up to Chapter 6 and presents the use of RSO predictions in anti-sun pointing as part of a larger SSA operations loop starting with predictions and ending with real-image validation. A pipeline like this one will be extremely beneficial for future SSA missions as we can better build a photometric payload with more advanced prediction and analysis tools.

In comparison to light curve analysis presented in Part I of this dissertation, the work described in this part has the following differences:

1. Represents an industry application
2. Studies space-based observation
3. Demonstrates one part of performing astrometric analysis with images

The contributions of this part are as follows:

1. Developed an analysis pipeline for optical camera detectability analysis for co-orbiting objects which includes the analysis of host-target locations as well as brightness of different cameras.
2. Proved the viability of using simulations to predict target locations ahead of time implemented a specified pointing mode to enable RSO simulations.

While the work has been published in collaboration with industry partners, the contributions presented above are the sole work of Randa Qashoa. Research partners specified the mission parameters for the Redwing mission and provided the specific image timestamps for these studies but the methodology is the original work of Randa Qashoa.

Chapter 5

Wide FOV Imagers for Co-Orbiting Object Detection

Abstract

Space-based optical systems are powerful tools for providing SSA and Space Domain Awareness (SDA) as they are not limited by weather or by fixed geographic location. As part of Canada's new microsatellite SDA mission, named Redwing, a wide field imaging capability will be employed for both proximity and co-orbital awareness around Redwing's LEO. Repeated close-range observations of "co-orbiting" objects are highly valuable for accurate orbit determination, as well as assessment of collision risk for the host platform. Co-orbiting objects present a geometric challenge in observation, however, in that the look angles can vary rapidly during close approaches [108]. In addition to co-orbital objects, the increasing congestion in LEO is giving rise to many objects in near-coplanar orbits of different altitudes. A method to survey coplanar and co-orbital space objects is somewhat underdeveloped and observability issues are weakly addressed. As many optical space surveillance cameras' targets are often viewed at ranges greater than 1000 km, a technology and concept of operations gap exists when performing space surveillance on objects at ranges less than 250 km from the host camera in space.

A variety of different imaging options are utilized for RSO detection. To have an awareness of as many RSOs near the host platform as possible, as opposed to having a more detailed view of a smaller portion of the sky, wide FOV systems are used. These can be dedicated cameras for RSO detection, or they

could be cameras developed for other purposes, such as star trackers that are used in a dual-purpose fashion. Even though wide FOV cameras tend to have smaller apertures and therefore do not have the faint magnitude detection capability of large-aperture / narrow-FOV systems, the ability to study larger volumes of nearby sky is highly desirable for the co-orbital tracking application. There are in fact many examples of using wide FOV systems for SSA, including projects that the authors have previously participated in. For example, in [12] CASSIOPE's FAI has been used for RSO detection, while in [11], the PCO Panda 4.2 and the UI-3370CP-M-GL IDS were used to capture starfield images from a stratospheric balloon platform at about 40 km in altitude.

The goal of this study is to compare different small-aperture / wide FOV cameras by assessing their RSO detection limits. This includes finding the number of detectable co-orbiting RSOs that regularly pass within 250 km of the observer, the expected brightness of detected RSOs, as well as other supporting statistics. When designing an SSA mission, choosing the correct camera is a challenging task but one with important consequences on achieving mission goals. By comparing the performance of different cameras in RSO detection performance, we can determine which cameras would best accomplish mission objectives.

To perform this analysis, an algorithm was developed to model the orbit of the observing platform along with orbits of potential nearby RSOs in the sky around it. Then the brightness and Signal-to-Noise Ratio (SNR) for different cameras are calculated to determine the detection capabilities for these cameras. For this study, a comparison between three cameras is made. The cameras chosen for the comparison are the FAI, the PCO Panda 4.2, and UI-3370CP-M-GL IDS cameras. All three of these cameras have been used for SSA purposes in previous research, and have well-established optical characteristics, which make them ideal candidates for this study. A crucial extension to this, which is examined at a preliminary level, is the use of repeat observations by these wide FOV cameras during close approaches to obtain useful orbit knowledge.

This analysis is being performed with an eye to using low-cost, off-the-shelf, and dual-purposed cameras to provide SSA across multiple distributed platforms, including non-SSA missions. The authors are performing proximity analysis for the Redwing mission, so this research is being applied to dedicated SSA/SDA mission development as well as being a general camera trade study.

5.1 Introduction

Within the space community, there has recently been a growing interest in the fields of SSA and SDA. SSA and SDA focus on understanding RSOs, such as satellites, rocket bodies, and debris that are orbiting the Earth by detecting, tracking, and identifying the RSOs orbiting near our satellites of interest. There are many different approaches to tackle SSA and SDA and one of the most promising ones is developing SSA satelllites with optical systems. In our previous studies, we determined the feasibility of using low spatial resolution wide FOV cameras, such as star trackers, for RSO detection. As a technology demonstration for a future SSA mission, we launched two payloads on stratospheric balloons in 2022 and 2023 to test the capability of different cameras for RSO detection. As demonstrated in [11], the UI-3370CP-M-GL IDS and the PCO Panda 4.2 (in [11, 109]) were able to successfully capture RSOs under near-space observation environments. Additionally, in [12], we demonstrated the use of the FAI which is a camera onboard the CASSIOPE for RSO detection. Given that we verified that all three cameras are capable of capturing RSOs, we chose these cameras for the analysis presented in this study. The main camera parameters for the FAI, PCO, and IDS cameras are provided in Table 5.1.

When designing a space mission, selecting the appropriate payload to mission objectives is of great importance. When it comes to SSA missions where the main function of the optical camera is to capture as many nearby RSOs as possible, it becomes difficult to assess which camera would be sufficient for the mission. Most star tracker datasheets do not provide direct values for SNR and visual magnitude detectability limits. As a result, there is a need to properly analyze the available parameters of different SSA cameras to determine the capability of each camera in co-orbiting RSO detection. This study demonstrates a detectability analysis that is used to compare and determine the capabilities of the FAI, PCO, and IDS cameras to observe nearby co-orbiting objects. We are currently working on Canada's next generation SSA mission, named Redwing, and are implementing a similar analysis for selecting the appropriate hardware for the mission.

5.2 Methodology

In this section, we explore the parameters used in defining the study as well as an explanation of the SNR calculation needed to determine the detectability of objects.

Table 5.1: Camera parameter comparison between the FAI, PCO, and IDS cameras.

Name	Type	Spatial Resolution	Pixel Pitch (µm)	Focal Length (m)	FOV (degrees)	Average Angular Resolution Per Pixel (deg/pixel)	Aperture Diameter (mm)
FAI	Charge-Coupled Device (CCD)	256 x 256	26	0.0136	26	0.102	17.25
PCO Panda 4.2	Scientific Complementary Metal Oxide Semiconductor (sCMOS)	2048 x 2048	6.5	0.025	29.6	0.014	57
UI-3370CP-M-GL IDS	Complementary Metal Oxide Semiconductor (CMOS)	2048 x 2048	5.5	0.016	41	0.02	15.93

5.2.1 Study Parameters

For this analysis, multiple parameters are defined that are adjusted to accommodate different mission scenarios. Firstly, the orbit of the host or observer RSO is defined. In this study, we are using CASSIOPE's orbital parameters on March 21, 2023 as obtained by its Two-Line Elements (TLEs). In this study, as well as for the Redwing microsatellite mission, we are focusing on co-orbiting objects given the challenge in observing them due to the quickly changing look angles that occur due to the difficulty of using angles-only measurements in proximity operations as mentioned in [108]. These measurements also do not have enough parallax for accurate range determination, which makes orbital estimation more challenging. The orbital parameters for the host as well as the variation range of each parameter for a co-orbiting object are provided in Table 5.2. The parameters were varied according to a normal distribution.

Table 5.2: Host and target orbital parameters.

Orbit Element	Host (CASSIOPE) Orbit	three-Sigma Variation Relative To Host
Semimajor Axis	7115.7 km	$\pm$ 100 km
Eccentricity	0.0584161	$\leq$ 0.015
Inclination	80.9642°	$\pm$ 1°
RAAN	332.1°	$\pm$3°
Argument of Perigee	213.1°	0° to 360°
True Anomaly	147.7°	0° to 360°

When performing alternative studies, the amount of variation to define a co-orbiting object does not need to be adjusted unless the host orbit has a large eccentricity. The only change would be in the host orbital parameters. Following the orbital parameters, the other parameters that impact the study are the duration of the study window, the threshold distance to assign an RSO as a co-orbiting object, the SNR threshold for detectability, and the diameter of the target object. The values of these parameters that are utilized in this study are in Table 5.3.

As per Table 5.3, the orbit of the host and the surrounding targets are propagated for one day. While using the orbital parameter range in Table 5.2 generates co-orbiting objects, this study focuses on very close approaching targets within 250 km of the observer RSO. This threshold was chosen to focus this study on nearby object surveillance. Given that we are looking for the detectability limits of each

Table 5.3: List of detectability analysis parameters.

Analysis Parameters	Value
Orbit propagation duration	1 day
Number of generated targets	100
Maximum host to target distance	250 km
Minimum SNR	6
Diameter of target RSO	20 cm
Background visual magnitude	20 per square arcsec

camera, the study focuses on small 20 cm diameter targets such as debris and cubesats. A minimum SNR of six allows us to ensure a true detection. All these parameters are interchangeable for future mission studies where the analysis requirements differ from this study.

5.2.2 Determining Camera Detectability

This section explains the steps taken to determine the number of detected objects for each camera. The first step is to calculate the position vector for the host using the orbital parameters in Table 5.2 for the full propagation duration, which is set to one day in this study. The next step is to generate the orbits for the 100 surrounding targets. A normal distribution is implemented where the mean is the original host's orbital parameters and three standard deviations is the variation relative to host as per Table 5.2. The only exceptions to this are the argument of perigee and the true anomaly which are randomly assigned values between zero and 360 degrees. Using the co-orbiting orbital parameters, a set of position vectors are generated for each of the 100 targets. Following that, the host to target distance is calculated and all objects further than 250 km are excluded from the rest of the study. The list of target position vectors and distances to host spacecraft is then saved to ensure consistency when testing the different camera parameters.

With the orbital information known, the detectability analysis can be performed. There are two different analyses performed. The first analysis is to determine the impact of integration time on the number of detected RSOs from the selected cameras: FAI, PCO, and IDS. The second, more detailed analysis focuses on the value of object visual magnitude and SNR at the target RSO's entry, exit, minimum rate, and minimum distance points. Regardless of which analysis is performed, the main values to calculate are the visual magnitude and SNR. Equation 5.1 based on the equation in [110] is used to calculate the solar phase angle and Equations 5.2- 5.3 from [111] are used to calculate the visual magnitude.

$$\Phi = \pi - cos^{-1}([\mathbf{r_{target-ECI}} - \mathbf{r_{host-ECI}}] \cdot \mathbf{r_{sun-ECI}}) \tag{5.1}$$

$$F(\Phi) = \left(\frac{2}{3\pi^2}\right)[(\pi - \Phi)cos\Phi + sin\Phi] \tag{5.2}$$

$$m_{obj} = m_{sun} - 2.5log\left(\frac{\rho A_{target} F(\Phi)}{R^2}\right) \tag{5.3}$$

In Equations 5.1 - 5.3, Φ is the solar phase angle which is the angle between the direction of the target to the sun and the target to the host RSO as shown in Figure 5.1 from [110]. The phase angle is calculated using the dot product of the target position as measured from the observer in the Earth Centered Inertial (ECI) frame and the sun's ECI position vector. The π factor is included due to the position being described in the host frame. While phase angle was originally used in astronomy for calculating values for objects at far distances, a similar formulation is applied to RSOs in which the target and host are at much smaller distances to each other. $F(\Phi)$ is a function relating the illumination phase angle to the fraction of light reflected. This function is typically a combination of specular and diffuse reflection. Given that the focus of this study is on the position of RSOs, a spherical RSO has been assumed and through testing we have determined that specular reflection results in negligible impact to the SNR so only diffuse reflection is applied which results in Equation 5.2 from [111]. If the study was focused on finding the best camera for a photometric mission, a more in-depth visual magnitude calculation would have been included. m_{obj} is the visual magnitude of the target and m_{sun} is the visual magnitude of the sun which is -26.74. ρ is the target RSO's diffuse reflectivity which is assumed to be 0.2 in this study as per [31] . A_{target} is the area of the target which is $\pi d_{target}^2/4$ for a spherical object where d_{target} is the target RSO's diameter. Lastly, R is the range or the distance between the target and the RSO.

Using the visual magnitude, the number of signal photoelectrons are calculated using the equations presented in [112] which are reproduced in Equations 5.4 and 5.5.

Figure 5.1: Phase Angle Diagram. Source: [110]

$$E_{RSO} = 5.6 \times 10^{10} \times 10^{-0.4m_{obj}} \tag{5.4}$$

$$e_s = Q_e \times \tau \times A \times E_{rso} \times t_{signal} \tag{5.5}$$

In Equations 5.4 and 5.5, E_{RSO} is the photon irradiance in $ph/s/m^2$, Q_e is the camera's quantum efficiency, τ is the optical transmittance, A is the aperture area which is calculated as $\frac{\pi D^2_{aperture}}{4}$. The equation for signal provided in [112] also adds an atmospheric transmittance factor but this is assigned to be one in this study since we are assessing a space-based observer as opposed to a ground-based observer. t_{signal} refers to the signal integration time which has a maximum value equal to the camera's selected integration time. In practice, integration time is usually less than this maximum value as a target with an angular rate relative to the camera will be within the camera's FOV for a limited period. The system integration time is calculated using Equations 5.6 and 5.7.

$$\theta_{pixel} = \frac{PixelPitch}{FocalLength} \tag{5.6}$$

$$t_{signal} = minimum\left(\frac{\theta_{pixel}}{\omega}, t\right) \tag{5.7}$$

In Equations 5.6 and 5.7, θ_{pixel} is the instantaneous FOV which is calculated using a camera's pixel pitch and focal length. ω is the angular rate of the target and t is the camera integration time.

With the signal value calculated, the remaining value needed for the SNR is the background noise. Equations 5.8 and 5.9 from [112] are used to calculate the background noise.

Table 5.4: Noise parameter comparison between the FAI, PCO, and IDS cameras.

Name	$e_{DC}(electron/pixel/second)$	$e_n(electron/pixel)$
FAI	529	10
PCO Panda 4.2	15	2.3
UI-3370CP-M-GL IDS	3	30

$$L_b = 5.6 \times 10^{10} \times 10^{-0.4M_b} \times \left(\frac{180}{\pi}\right)^2 \times 3600^2 \tag{5.8}$$

$$e_b = Q_e \times \tau \times L_b \times A \times \theta_{pixel}^2 \times t \tag{5.9}$$

In Equations 5.8 and 5.9, L_b is the background signature in $ph/sec/m^3/sr$, M_b is the background visual magnitude per square arcsec. The value of M_b includes sky brightness as well as stray light from any other sources that would impact the analysis, and e_b is the number of background photoelectrons. The values for e_b is the same across all timesteps for each camera in this study. With the signal and noise functions, the SNR is calculated as per Equation 5.10 based on the equation in [112].

$$SNR = \frac{e_s}{\sqrt{e_b + e_n^2 + (e_{DC} \times t)}} \tag{5.10}$$

In calculating the SNR in Equation 5.10, three sources of noise are considered: the background noise e_b, read noise e_n, and the dark current noise e_{DC}. In [112], dark current is not taken into consideration. We have taken dark current into consideration in this study due to studying a space-based observer, which encounters more extreme temperatures, as opposed to a ground-based observer. The values assumed for each camera for e_n and e_{DC} are in Table 5.4.

5.3 Results

This section shows the results of performing the analysis as explained in Section 5.2. The first step is to propagate the host's (CASSIOPE's) orbit and to generate 100 random co-orbiting objects as per the orbital parameters listed in Table 5.2. These parameters were varied according to a normal distribution, with the values in Table 5.2 representing the mean, in the first column, and a three-sigma variation, in the second column. Out of 100 randomly distributed targets, 13 RSOs have instances where they are within 250 km of the observer with a total of 25 close approaches for a span of one day. Fig. 5.2 shows a semi-log plot of the host to target distance over the one day scenario for the 100 generated targets.

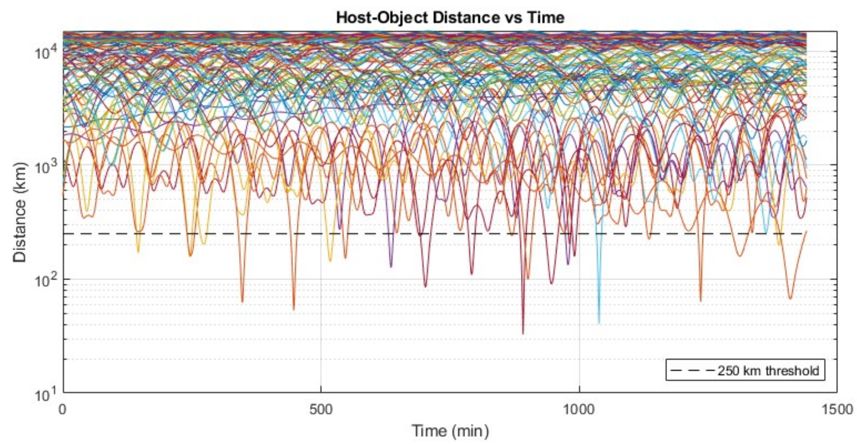


Figure 5.2: Plot of host to target distance for 100 randomly generated co-orbiting objects for one day.

Fig. 5.3 shows the directions of the close approaching targets with regards to the host's orbit. From Fig. 5.3, most close approaching targets are near the host orbit's normal or anti-normal vector. If we are planning a mission and deciding on where to aim the camera to best detect nearby RSOs, in this particular case, aiming it such that the FOV is along the orbit's normal or 180 degrees from the orbit normal vector ensures that most nearby objects would be visible. This is only concerning the geometry of the orbits as we have not yet assessed the detection capabilities of the different cameras.

With the host to target orbits understood, the next part is to determine how well each of our three cameras, the FAI, PCO, and IDS are capable of detecting nearby RSOs. Most cameras allow the user to choose from a range of different integration times. Fig. 5.4 shows the impact of integration time on the number of detected RSO sequences for each of the three cameras. As noted in Table 5.3, an RSO is considered to be detected if it is observed with a minimum SNR of six.

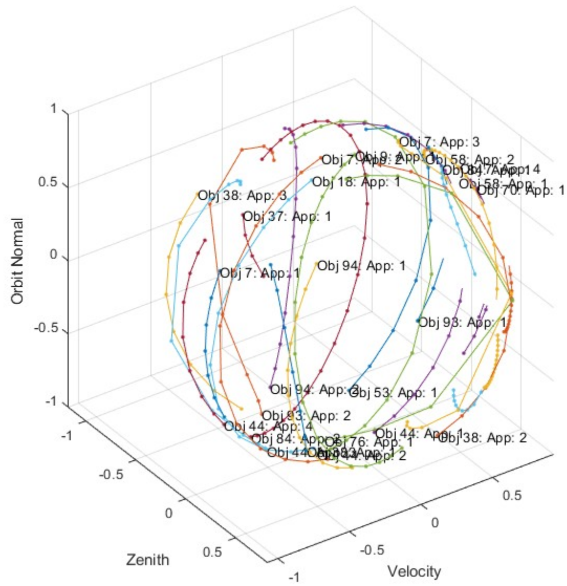


Figure 5.3: Directions of close approaching targets within the host's orbit.

From Fig. 5.4, we determine that the PCO performs best for detection followed by the FAI and then the IDS camera. For the PCO camera, the number of detected close approaches decreases as the integration time increases with a maximum detection rate for integration times less than or equal 500 milliseconds. For the FAI, the number of detected objects first increases and then decreases with integration time with a maximum detection rate at integration times of 700 - 800 milliseconds. For the IDS camera, the detection rate remains stable until an integration time of about 5.5 seconds where it begins to decrease. The impact of integration time is different for the different cameras due to a combination of two factors. First, an increase in integration time increases the number of received photons in the detector which would increase the SNR but this effect is countered by the fact that the increase in integration time increases the length of the RSO streak. This in turn increases the impact of noise that subsequently reduces the SNR. In most cases, the streaking causes a reduction in the SNR with the exception of FAI for integration times up to 800 milliseconds where the increasing photon count supersedes the impact of the additional noise.

The last analysis considers the distance, visual magnitude and SNR for the close approaching targets at four points in each sequence: the entry, exit, minimum rate, and minimum distance points. The entry and exit points are studied as they represent the beginning an end of each sequence. The point

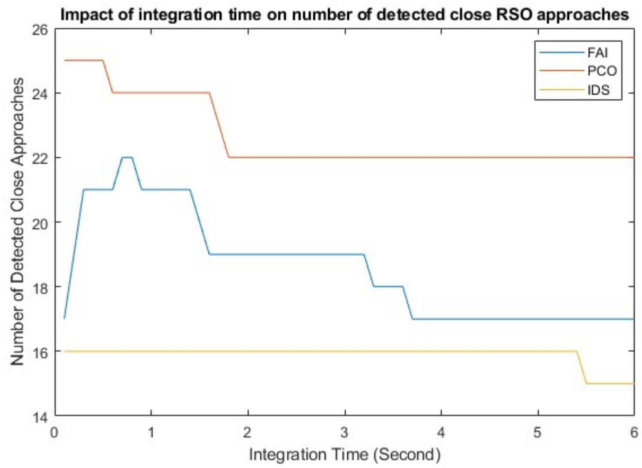


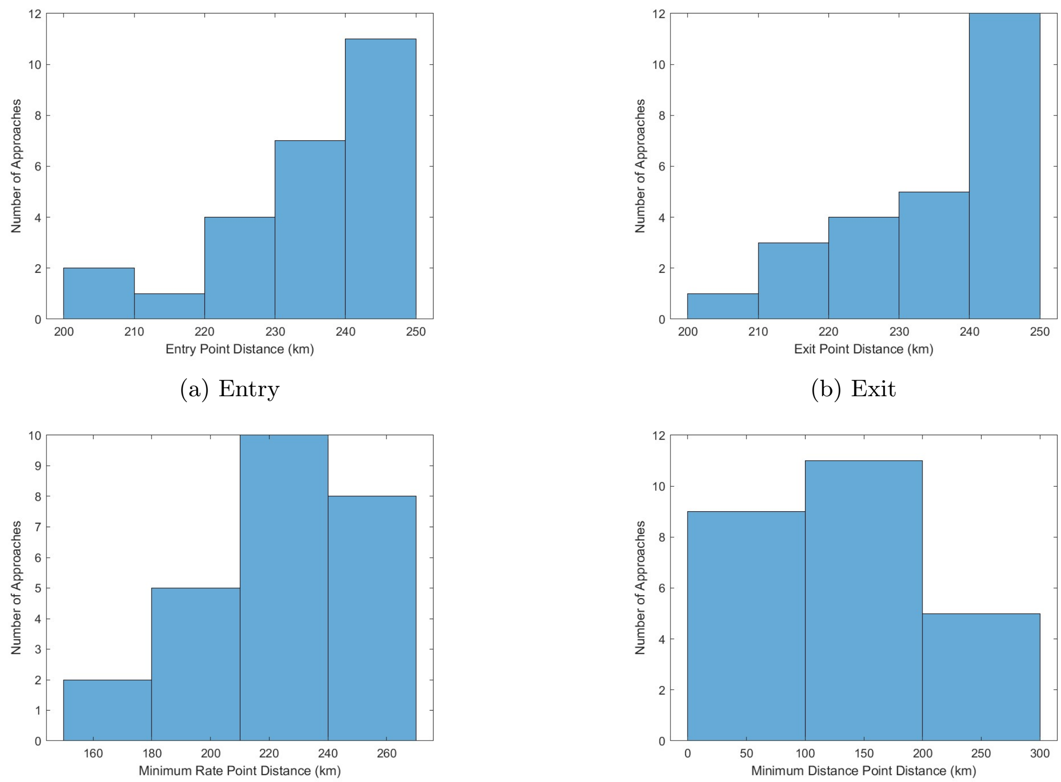
Figure 5.4: Impact of integration time on the number of detected close approaches for FAI, PCO, and IDS.

of minimum angular rate is studied due to the reduced host to target relative motion. This means the target may be detected more easily. Lastly, the minimum distance point is studied as the reduced distance is expected to increase the brightness of the observed target. Tables for all values used in the analysis are in Appendix A. Figure 5.5 shows a histogram of the host to target distance for every close approach.

While the distances at the entry, exit, and minimum rate points are close to 250 km, the minimum distance is typically much smaller. It is less than 100 km in multiple instances. Figure 5.6 shows the histograms of the visual magnitude for each target's approach at the entry, exit, minimum rate, and minimum distance points.

The exit point has the lowest number of bright objects while the minimum distance point has the largest number of bright objects. To determine how well target approaches are observed from the host RSO, Figures 5.7 - 5.9 show the SNR histograms for each camera at an integration time of 100 milliseconds.

The first bin in all the SNR figures is for the non-detectable objects with SNR less than six, The best SNR results occur when the PCO camera is used for observation followed by the FAI and IDS cameras. The large aperture and small pixel pitch allows the PCO to detect RSOs with better quality than the other two cameras.



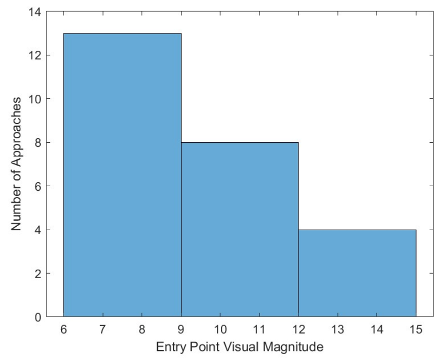
(a) Entry (b) Exit

(c) Minimum Rate (d) Minimum Distance

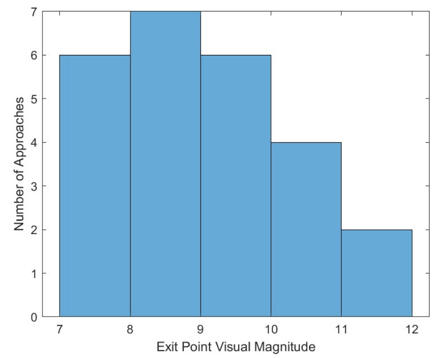
Figure 5.5: Histogram of host to target distances

5.4 Conclusions

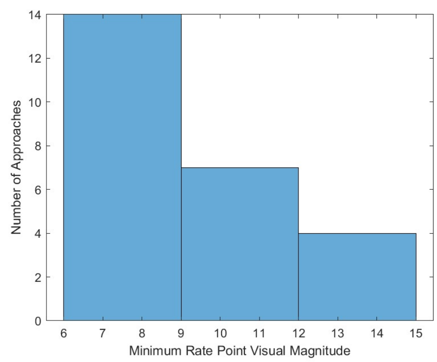
In conclusion, this study presents a pre-mission detectability analysis to allow mission developers to assess the capabilities of different cameras for a particular mission. This analysis studied the orbit of the CASSIOPE satellite to compare the performance of three cameras in detecting small and nearby RSOs. The analysis considered multiple factors which are the camera orientation, integration time, and the SNR of the targets. Using the directions of potential close approaches, we determined that aligning the camera with the orbit normal or anti-normal frame would result in the most detections. The plot of target entry directions highlights the importance of choosing the correct camera orientation as not all orientations are as densely populated with targets as the normal or anti-normal frames. When looking at integration times, the PCO camera outperformed the FAI and IDS in the number of detected RSOs. The best performance of the PCO is at integration times below 500 milliseconds. When comparing the SNR of different targets, the PCO camera again outperforms the other two cameras. This analysis



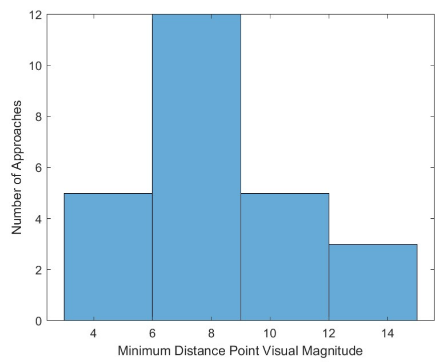
(a) Entry



(b) Exit



(c) Minimum Rate

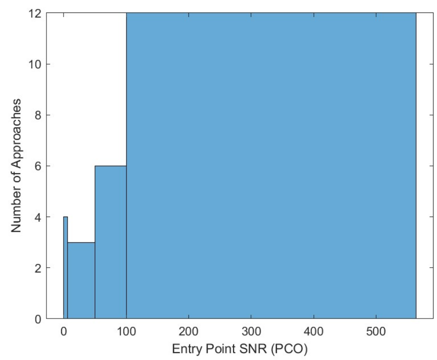


(d) Minimum Distance

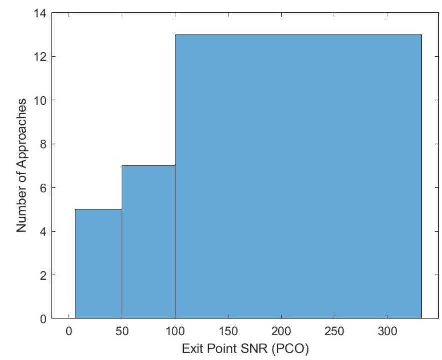
Figure 5.6: Histogram of target visual magnitudes

can be extended to other missions using the provided equations. Future work includes combining this analysis tool with TLEs of actual targets. At the moment, a series of 100 randomly generated co-orbiting objects was generated. This is to be expanded to add the option for propagating all RSOs with available TLEs to determine which RSOs would truly be co-orbiting to determine how often it happens and if there are any other trends of interest.

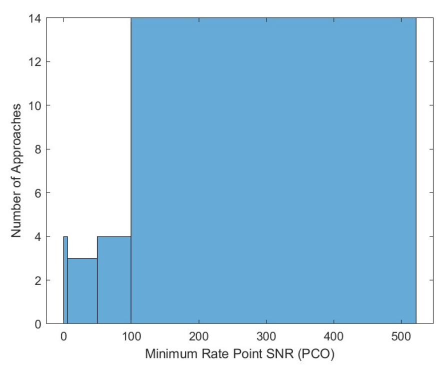
While this study focused on SNR as the metric for finding the optimal camera, future studies can look into optimize one or more different parameters such as phase angle, average duration of observation, brightness, etc. SNR was chosen in this study to best align with the purpose of this hypothetical mission, which is to observe co-orbiting RSOs within 250 km.



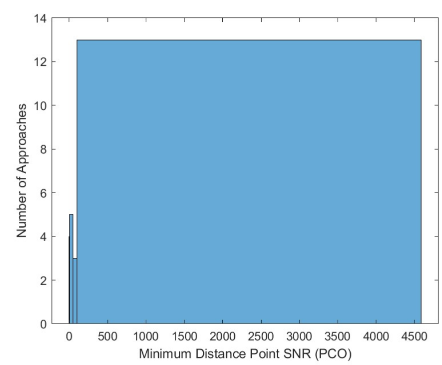
(a) Entry



(b) Exit

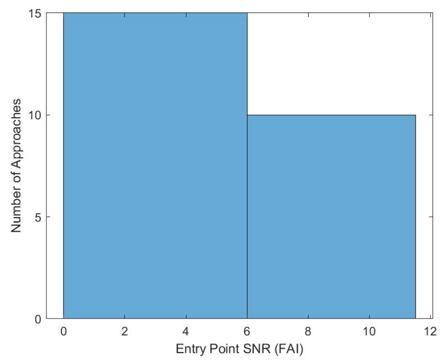


(c) Minimum Rate

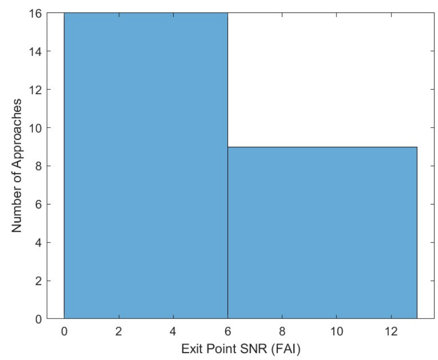


(d) Minimum Distance

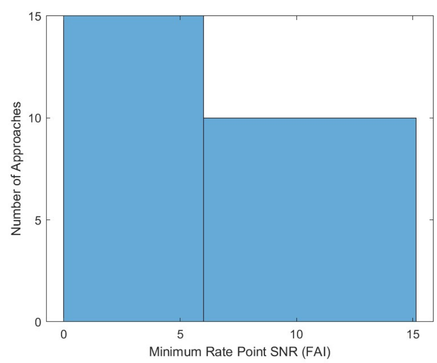
Figure 5.7: Histogram of PCO SNR values



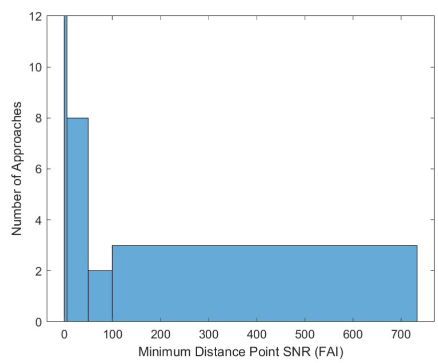
(a) Entry



(b) Exit

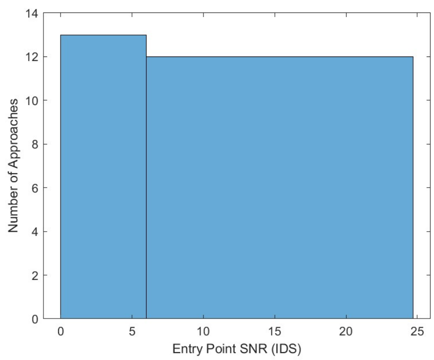


(c) Minimum Rate

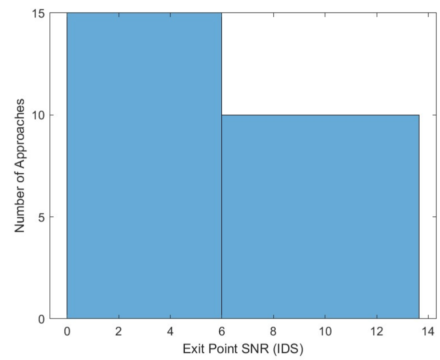


(d) Minimum Distance

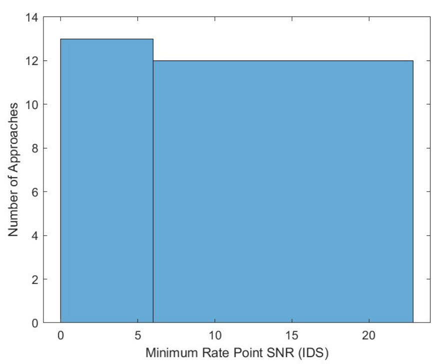
Figure 5.8: Histogram of FAI SNR values



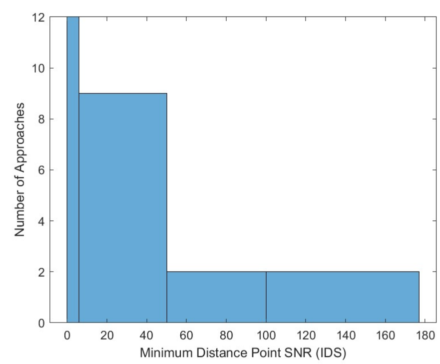
(a) Entry



(b) Exit



(c) Minimum Rate



(d) Minimum Distance

Figure 5.9: Histogram of IDS SNR values

Chapter 6

RSO Simulations with Anti-Sun Pointing Predictions

Abstract

A major component of SSA is the ability to predict, track and identify RSOs in orbit. High fidelity RSO simulators are a critical tool in developing RSO tracking and detection algorithms to generate simulated star field images with accurate representation of RSOs. RSO characteristics such as position, velocity, etc. are calculated from TLEs, Orbital Mean-Elements Message (OMM), or ephemeris data. Most RSO simulators used in today's SSA studies (including [31], developed in our research team) lack pre-built attitude profiles of the host observer, which is needed to determine the camera's boresight, or pointing direction. In this study, we present improvements to the simulator to include a wider range of inputs for the attitude information. Specifically, by introducing a known pointing profile for the satellite we can make more accurate predictions of the RSO's position and velocity.

In exploring the addition of a known pointing profile to the simulator in cases where attitude information is lacking, we additionally limit ourselves to wide FOV and low spatial resolution cameras, akin to the ones used in star trackers, although the simulator can support high spatial resolution image generation as well. We limit the study to wide FOV cameras due to their ability to detect more RSOs. Images from CASSIOPE's FAI camera were chosen due to similar camera parameters to star trackers and the

availability of public ephemeris files that we can use to generate even more realistic simulations for comparison. With the ephemeris and attitude information, it is possible to extract the exact boresight vector for the camera for every timestamp being studied. The FAI is currently locked in an anti-sun pointing mode due to a recent issue with CASSIOPE's reaction wheels [113] making it an ideal candidate for this study. An anti-sun pointing mode was implemented within the simulator and simulated images from ephemeris files will be used as validation data.

The validation will be accomplished by simulating FAI images that were predicted using the anti-sun pointing mode with simulated images generated using true ephemeris files that contain attitude information. In comparing the images, we will look at specific targets' temporal features. Some of the temporal features we will compare are the entry and exit times between both sets of images as well as the observation duration between both methods. By proving the ability to predict a certain, known, spacecraft orientation with the simulator, we can then further improve its prediction capabilities by adding more orientation modes.

Many satellites utilize specific orientation profiles to achieve their various mission goals. Such pointing modes include anti-sun, nadir, or inertial pointing. Most satellites have known pointing modes that they reside in most of the time. However, they do not have publicly available attitude information, which impacts our ability to simulate images of star fields and RSOs. Our proposed solution is to establish general classes for the host spacecraft pointing rules to approximate the attitude of the spacecraft so we can generate more accurate simulated images in cases where we are limited by the insufficient amount of data.

6.1 Introduction

An important part of SSA and mitigating the concern of the increase in the number of RSOs is to detect and track the RSOs with accurate orbit and attitude information of the objects. This tool helps address the concern for the proliferation of RSOs which inevitably increases the risk of collisions for satellites. Thus, to provide timely and accurate information to enable safe and sustainable space operations, the tracking and identification of RSOs is of the utmost importance [114]. There are many ways to accomplish the tracking and identification of RSOs. One approach is taking images of star fields and identifying RSOs within them and looking at the positions of the various objects over multiple frames of an image sequence. Using this information and the time at which the object leaves the observer's FOV, one can correlate the object to a catalog number and/or determine some astrometric properties. The goal is to generate point source images for astrometric analysis in which the location of the RSOs and the distance to the host are taken into account. Unlike photometric studies, the shape and size of the RSO is not significant in astrometric analysis. However, in order to develop and test such algorithms, many validation images with known target RSOs are needed, but are scarcely publicly available with labeled data sets.

In order to develop algorithms for RSO detection and tracking, often large datasets (labeled and annotated with orbit and attitude information) are required for training and accuracy validation. This is especially the case if we need to develop ML algorithms. Given the difficulty of obtaining the data needed for training such algorithms, image simulations can be used for algorithm training. Simulations can easily be used to generate large quantities of labeled data with a lot of options for obtaining a variety of data by modifying one or more parameters within these simulators. Lack of labeled data is of the greatest challenge as in many applications and RSO images are no exception. There is no publicly available annotated data that can be easily used for algorithm design. Furthermore, manual labeling is extremely difficult given the nature of these images. For example, medical images are considered difficult because of the need for experts to label them. Cityscapes are challenging to annotate because of the large amount of subjectivity that could result from placing incorrect bounding boxes (or those of the wrong size) or even missing some labels when some smaller objects are neglected. Similarly, differentiating RSOs from stars and detecting them requires sequences of images instead of a single image, pre-processing to remove a lot of the noise to be able to see the potential RSOs and prior experience to be able to properly label RSOs.

Simulated images are often used in place of labeled real RSO images to provide accurate data. The accuracy of these images is critical for the development of analytical algorithms as we have truth data on the RSO and is helpful for training ML algorithms that perform well. Using simulated data to train ML algorithms is not an uncommon practice in this challenging field. For example, in [55], simulated light curves are used to train a ML algorithm for RSO light curve classification and in [115], simulated RSO images are used for RSO streak detection. By increasing the accuracy of RSO and starfield image simulators, we ease the process of transitioning from a fully simulated dataset to testing on real datasets.

In previous research, members of the Nanosatellite Research Lab have developed a starfield image simulator in [31] to help address the need for simulated datasets. The observer's position, velocity, and attitude can be determined by supplying measured values at small timesteps or by using a TLE file. For the purpose of generating realistic and accurate images, ephemeris files are preferred as they provide true values at almost constant instances (typically one second apart) as opposed to TLEs which must be propagated to the time of interest. One way to use TLEs when attitude files are unavailable is by specifying the orientation profile of the RSO as TLEs store information regarding the orbit of the RSO which can be used to extract position and velocity with no way to determine attitude. The only other method of using TLEs when attitude is unavailable is to generate tracking mode images where the camera tracks a specified target such that it is always in the center of the image. By providing rough estimates of the RSO's attitude profile, we can obtain simulated images in cases that were not possible before. Many RSOs have typical pointing modes to accomplish different objectives such as anti-sun pointing, nadir, or inertial pointing. By implementing such attitude modes within the pre-existing RSO simulator, we can improve the fidelity even when we do not have detailed attitude files. In this paper, we present the implementation of anti-sun pointing mode to accompany TLEs for RSO simulations when detailed attitude information is unavailable. The results presented are those with pointing mode operation of the CASSIOPE satellite particularly the FAI that is currently locked in a roughly anti-sun pointing mode. CASSIOPE has publicly available ephemeris data containing measured position and attitude which will be used to test the results obtained by using a combination of TLEs and anti-sun pointing mode. When using the ephemeris data, the position of CASSIOPE at the non-measured timesteps will be obtained using clamped cubic spline interpolation.

In Section 6.2, we describe the CASSIOPE and RSO simulator in more detail. In Section 6.3, the simulation sequences are discussed, followed by methodology of implementing anti-sun pointing and the simulation approach in Section 7.3. The results are shown and discussed in section 6.5 and the final remarks with the future tasks are provided in Section 6.6.

6.2 Background

6.2.1 CASSIOPE

CASSIOPE is a Canadian space weather satellite with multiple cameras. One of the main payloads is the FAI which was designed to study the aurora in the 650-1100 nm wavelength scale [86]. Recently, we have used FAI images to demonstrate that wide FOV cameras like star trackers and FAI imagers are capable of imaging RSOs for SSA applications [116]. The camera specifications of FAI are comparable to a star tracker, which was the main focus of SBOIS, in that it has a low spatial resolution of 256x256. The reason for focusing on low spatial resolution and wide FOV cameras such as star trackers for SSA is that we already have a large number of such cameras in orbit accomplishing different tasks. By using these cameras for SSA, we can utilize already existing hardware in space for newfound purposes such as RSO tracking, detection, and identification. One of the main advantages of using CASSIOPE as the observer is that the downlinked are publicly available and are easy to access. This includes FAI images as well as ephemeris files that contain CASSIOPE's position, velocity, and attitude information. This makes CASSIOPE the ideal RSO to use for SSA algorithm development and validation due to the availability of real data. Once the feasibility of a certain technique has been proven using CASSIOPE, the algorithms can be extended for studies on other RSOs.

Prior to 2021, CASSIOPE was able to point at specified targets. Unfortunately, at the end of 2021, CASSIOPE's three of four reaction wheels ceased to function so the satellite is no longer able to point at targets [113]. Instead, the spacecraft is oriented such that its top solar panel is always facing the sun. Due to the location of FAI on the satellite bus, which is on the opposite end of the solar panel, it will point in an anti-sun direction which is ideal for RSO viewing.

6.2.2 RSO Simulator

The RSO simulator, referred to as SBOIS was initially developed by colleagues during a past feasibility study to determine whether star trackers can be used for RSO imaging [31]. FAI simulations from SBOIS have been compared with System Tool Kit's (STK's) Electro-Optical Infrared (EOIR) tool and real images. As demonstrated in [31], SBOIS is faster and more accurate than STK-EOIR. SBOIS has since been expanded to include many other features such as tracking mode implementation and ground-based imaging. We are continuously looking for methods to improve the simulator. In this chapter, we added and validated a specified pointing mode which is anti-sun pointing to SBOIS. Prior to adding anti-sun pointing, the only other method of performing simulations without ephemeris files required using TLEs with an attitude file which is typically not publicly available. Obtaining detailed attitude files are just as challenging as obtaining ephemeris files so by estimating the attitude as belonging to a certain profile (which applies to many RSOs), we can perform simulations on objects we were not able to before. By expanding the capability of the RSO simulator to generate images from any observer we were unable to simulate from previously, we added a larger variety of simulated images that can be used for ML training which potentially improves the performance of these algorithms. In [117], for example, simulated images were used for developing a classifier that differentiates stars, RSOs, and noise.

Simulated RSO data are also used in other areas of SSA research beyond AI-based detection algorithm design such as light curve analysis [118]. In addition to the simulator presented in this chapter, there have been several studies that proposed designs of RSO simulators such as [26] and [27].

6.3 Simulation Datasets

CASSIOPE has publicly available ephemeris files that can be downloaded from University of Calgary's enhanced Polar Outflow Probe (e-POP) website [119]. Data from the RSO's first moments of operations in 2013 till today are all available. We have simulated over 600 images using anti-sun pointing to test this new feature in the simulator. The comparison study described in this paper is performed on three images taken of RSOs in view of FAI on April 16 and April 22, 2023. These particular dates and time windows were chosen due to the availability of ephemeris files and real images on these days. The targets used for this analysis are represented in Table 6.1.

Table 6.1: List of Studied RSOs

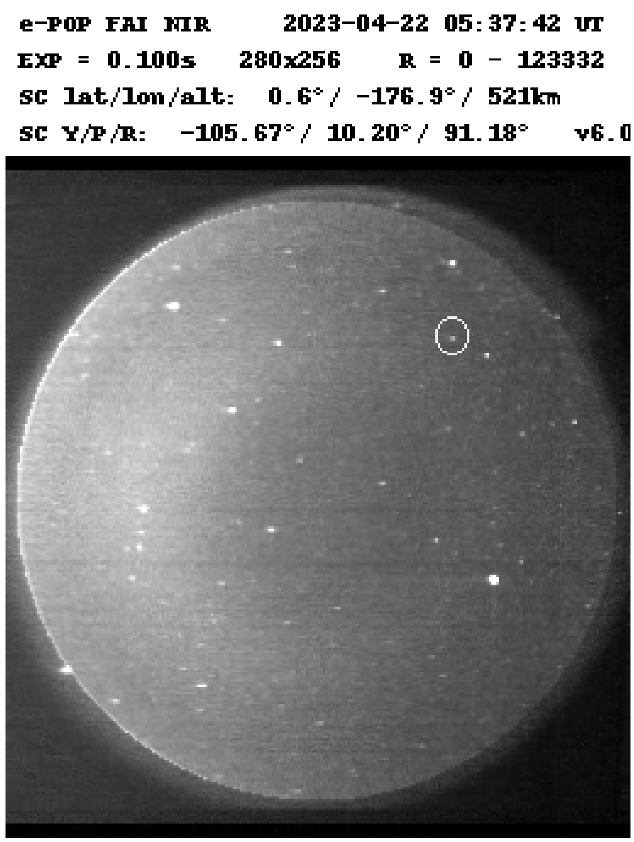
Dataset Number	**Target Name**	**Target Catalog Number**
1	USA 40 R/B DEB	23159
2	OneWeb-0647	56079
3	Starlink-6101	56115

USA 40 R/B DEB (catalog ID 23159) is a piece of debris from an American rocket body that was launched in 1989 [120]. At the time of observation, this rocket body was approximately 5756 km from CASSIOPE with a phase angle of around 18°.

The target in the second dataset is OneWeb-0647 which is a satellite that was launched on March 26, 2023 [121]. OneWeb satellites have dimensions of 1 m x 1 m x 1.3 m [122]. At the start time of the second dataset, this target was around 1935 km away from CASSIOPE and has a phase angle of about 24.5°.

Lastly, the third dataset targets Starlink-6101 which is another recently launched satellite on March 29, 2023 by SpaceX [123]. At around 23:50:25 UTC, CASSIOPE was able to see Starlink-6101 which was 605 km away with a 16.6° phase angle.

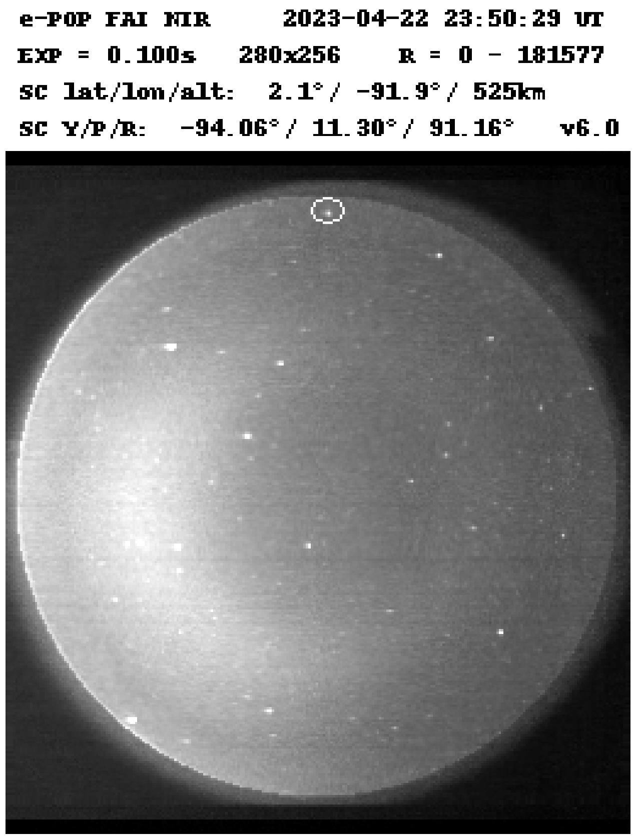
An image of the second target (OneWeb-0647) taken from FAI on April 22nd is shown in Fig. 6.1a and an image of the third target (Starlink-6101) captured on the same day is illustrated in Fig. 6.1b. Each of these images has multiple RSOs in view from FAI but, in this analysis, we will only focus on one target per image. The white circle indicates the location of the target RSO. This was determined using the expected location of the RSO from the ephemeris-based simulations.



(a) $Target : OneWeb - 0647$

Figure 6.1: Real FAI images captured at 5:37:42 and 23:50:29, respectively, UTC on April 22, 2023

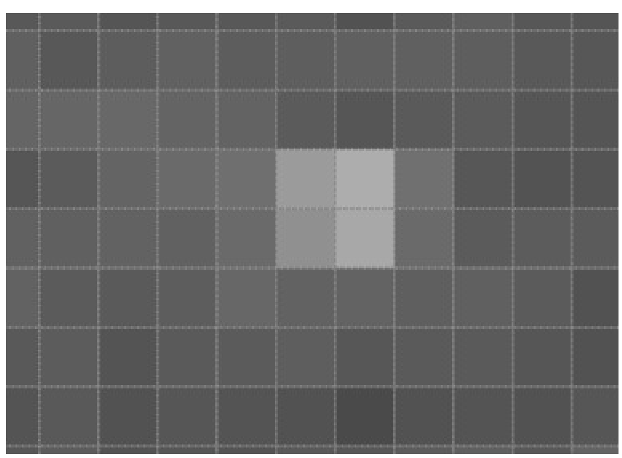
In close-up, images of the above targets appear to have four and five lit pixels, respectively as shown in Fig. 6.2. In this study, a pixel is considered to be a lit if its intensity is at least 60% of the maximum



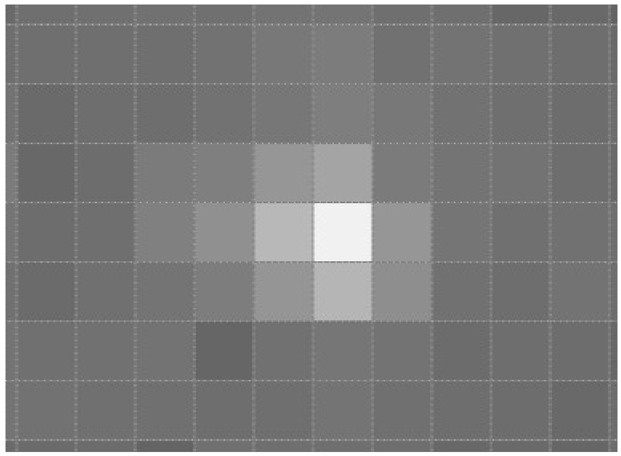
(b) $Target:\ Starlink-6101$

Figure 6.1: Real FAI images captured at 5:37:42 and 23:50:29, respectively, UTC on April 22, 2023

intensity in the image.



(a) $Target:\ OneWeb-0647\,(Four\,lit\,pixels)$



(b) $Target:\ Starlink - 6101\,(Five\,lit\,pixels)$

Figure 6.2: Lit Pixels of Targets from Real FAI Images

6.4 Methodology

6.4.1 Anti-Sun Pointing

The mathematical equations for determining the anti-sun pointing attitude are provided in equations 6.1-6.6. In order to orient the RSO in an anti-sun pointing direction, we need to first obtain the sun's position in the ECI frame. Then, we invert the sun's ECI coordinates to point anti-sun and convert to the host satellite's, which is FAI in this case, orbital frame. With the RSO in the orbital frame, we can convert the RSO's position vector to obtain the attitude needed to point to that vector, which is anti-sun in this case. To perform this conversion, we need the RSO's position in both the orbital frame as well as the body frame which we are assuming to be fixed on the z-axis. Using these two vectors, we need to calculate two matrices, which will be referred to as G and F for this study, and find the rotation matrix between them. The expression for G is shown in equation 6.1.

$$\mathbf{G} = \begin{bmatrix} \mathbf{r_{orb}} \cdot \mathbf{r_{bdy}} & -\|\mathbf{r_{orb}} \times \mathbf{r_{bdy}}\| & 0 \\ \|\mathbf{r_{orb}} \times \mathbf{r_{bdy}} & \mathbf{r_{orb}} \cdot \mathbf{r_{bdy}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.1}$$

The expression for the F matrix is in equation 6.2.

$$\mathbf{F} = \begin{bmatrix} \mathbf{r_{orb}} \\ \\ \dfrac{\mathbf{r_{bdy}} - (\mathbf{r_{orb}} \cdot \mathbf{r_{bdy}})\mathbf{r_{orb}}}{\|\mathbf{r_{bdy}} - (\mathbf{r_{orb}} \cdot \mathbf{r_{bdy}})\mathbf{r_{orb}}\|} \\ \\ \mathbf{r_{bdy}} \times \mathbf{r_{orb}} \end{bmatrix}^{\mathrm{T}} \tag{6.2}$$

Using both the G and F matrices, the rotation matrix can be calculated using equation 6.3.

$$\mathbf{R} = (\mathbf{FGF}^{-1})^{\mathrm{T}} \tag{6.3}$$

With the rotation matrix, we can calculate the yaw, pitch, and roll as shown in equations 6.4 - 6.6.

$$Yaw = \tan^{-1}\left(\frac{\mathbf{R}(2,1)}{\mathbf{R}(1,1)}\right) \tag{6.4}$$

$$Pitch = \tan^{-1}\left(\frac{-\mathbf{R}(3,1)}{\sqrt{\mathbf{R}(3,2)^2 + \mathbf{R}(3,3)^2}}\right) \tag{6.5}$$

$$Roll = \tan^{-1}\left(\frac{\mathbf{R}(3,2)}{\mathbf{R}(3,3)}\right) \tag{6.6}$$

A summary of the steps needed to determine the attitude of the RSO in anti-sun pointing is described in Fig: 6.3

For each time stamp, the simulator outputs an image as well as a data file containing information on the observed targets including their catalog number, distance to the host, and position in ECI.

6.4.2 Comparison Approach

For this study, we have decided to perform a temporal comparison between simulated images generated using ephemeris files, which are essentially truth data, as well as simulated images generated using TLEs in anti-sun pointing mode. We compared the time of entry and exit between the two approaches as well as the observation duration.

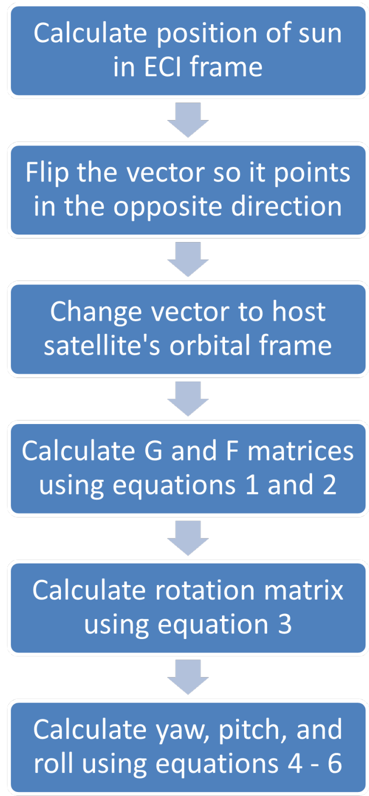


Figure 6.3: Flowchart for determining anti-sun pointing attitude

6.5 Results and Discussion

We generated simulated images of FAI using anti-sun pointing mode and with ephemeris files. To ensure we captured the full entry and exit times of all target RSOs, we generated simulations with a two minute buffer from the start times listed in Table 6.1. A sample simulated image from dataset three is depicted in Fig. 6.4. The artifact on the left of Fig. 6.4b is the Earth's limb. There is a slight difference in the boresight Right Ascension (RA) and Declination (Dec) such that the Earth limb does not appear in Fig. 6.4a.

Using the resulting data file that includes the information of the observed targets, we extracted the entry and exit times for each target. The results from the ephemeris file simulation are summarized in Table 6.2.

The next step in the study was to generate the same datasets but using CASSIOPE's TLE with anti-sun pointing instead of its ephemeris files. The results are summarized in Table 6.3.

By subtracting the value of start time and end time between the two methods, we obtain Table 6.4.

Table 6.2: Ephemeris File Dataset Entry and Exit Times

Dataset Number	**Catalog Number**	**Start Time (UTC)**	**End Time (UTC)**	**Duration (Seconds)**
1	23159	15:56:21	15:58:01	100
2	56079	05:37:37	05:38:00	23
3	56115	23:50:27	23:50:33	6

Table 6.3: Anti-Sun Pointing Dataset Entry and Exit Times

Dataset Number	**Target Catalog Number**	**Start Time (UTC)**	**End Time (UTC)**	**Duration (Seconds)**
1	23159	15:55:00	15:57:35	155
2	56079	05:37:01	05:38:10	69
3	56115	23:50:18	23:50:35	17

Negative values indicate that anti-sun pointing simulations start or end earlier, or are shorter than ephemeris file simulations.

Table 6.4: Difference Between Anti-Sun Pointing and Ephemeris File Dataset Entry and Exit Times in Seconds

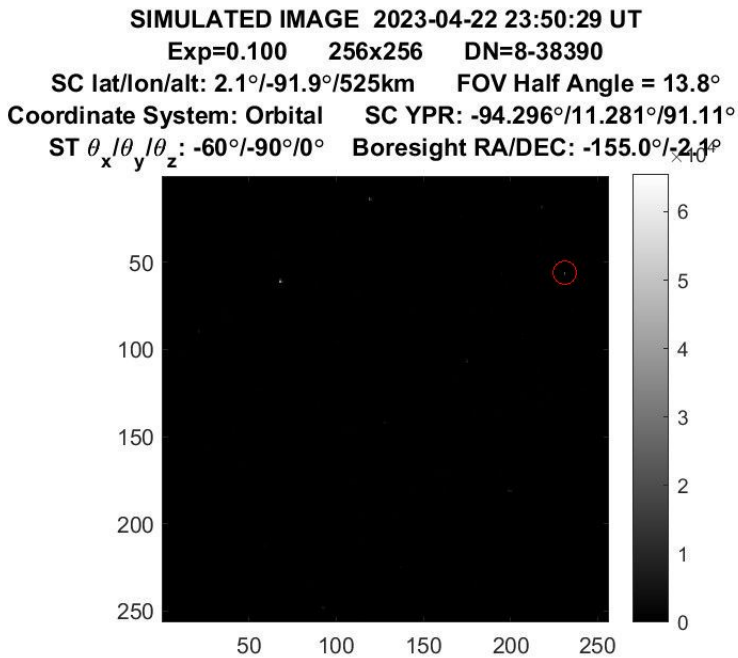
Dataset Number	**Target Catalog Number**	**Start Time Difference**	**End Time Difference**	**Duration Difference**
1	23159	-81	-26	55
2	56079	-36	10	46
3	56115	-9	-2	11

By comparing the two tables, we can notice that the same targets are still visible when using TLEs and anti-sun pointing mode but there is a difference in the entry and exit times compared to using ephemeris files only. The largest difference is 81 seconds. Given that observation windows for RSOs typically span several minutes, the difference is negligible for scheduling purposes or early detection which indicates that anti-sun pointing is still a viable method when measured attitude data are unavailable. In these examples, using anti-sun pointing mode, we can have earlier predictions of when a target RSO would be in view but the there is no specific trend on whether anti-sun pointing estimation predicts RSOs exiting earlier or later than when using ephemeris files. The fact that the same target RSO is in view of the camera using such a prediction within a minute and a half accuracy proves the viability of using such predictions when detailed attitude information is unavailable.

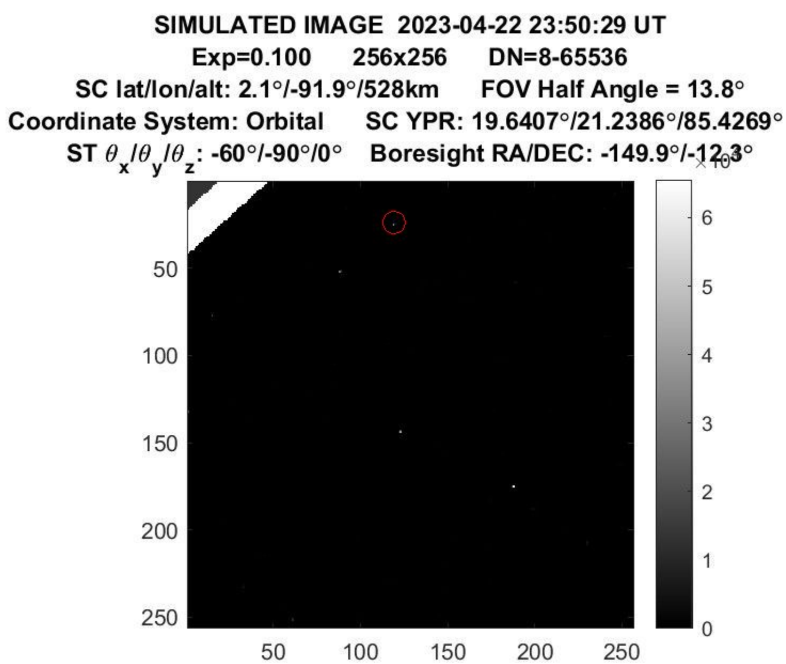
6.6 Conclusion

In conclusion, an improved simulator over [31] with the anti-sun pointing mode feature has been developed to perform simulations when detailed attitude information is not available, and the observer is known to be pointing in an anti-sun direction. This allows us to expand the number of simulated RSOs as ephemeris files are rarely made public and there is no other method of generating stare-mode simulated images. At all three datasets, we were able to see the specified targets with a maximum difference in entry and exit time of 81 seconds. The maximum difference occurs in the longest dataset and the difference drops down with shorter datasets. For a long dataset, the difference in entry time is less than two minutes so for estimates when ephemeris files are unavailable, we can use pre-built attitude profiles. The results presented here are of a limited initial test and we will find and test new datasets with anti-sun pointing simulations, ephemeris file simulations, and real images. The results in this paper are significant as we are able to accurately replicate images for mission planning and ML algorithm training for a wider variety of RSOs with the added attitude profiles.

Future work includes adding additional attitude profiles within the simulator, such as inertial pointing. Additionally, we can now use SBOIS to determine all targets in view of FAI even when measured attitude is not available. Future work also includes performing additional studies comparing the real and simulated images in order to locate uncataloged RSOs that do not have TLEs.



(a) *Generated Using Ephemeris File*



(b) *Generated Using Anti Sun Pointing Profile. The White Line On The Top Left Represents The Earths Limb*

Figure 6.4: Simulated Images Generated Using two Methods on April 15, 2023, 00:30:31 UTC

Chapter 7

Demonstration of RSO simulation, prediction and detection using a low spatial resolution imager in anti-sun pointing

Abstract

SSA is a field of increasing prevalence due to the large increase in the number of RSOs in orbit. Using low spatial resolution imagers, such as star trackers, for RSO imaging is a new and emerging field within SSA that is becoming more popular by the day. Despite the challenges of using low spatial resolution imagers for SSA, star trackers have wide FOVs which make them ideal for RSO monitoring. It also greatly enhances the distribution of observations as even smaller satellites such as cubesats are able to implement RSO detection and tracking capabilities using a star tracker alone.

To develop algorithms for RSO detection, tracking, identification, and even for photometric light curve extraction, RSO images are needed so the availability of these images is of the utmost importance. Given the difficulty in obtaining a large number of real images, due to downlink limitations and security

concerns regarding sharing RSO images, simulations are used to assist with such algorithm development. We have built such a simulator in [31]. Given that we do not always have an RSO's attitude information, we have implemented pre-defined orientation modes in our simulator. Additionally, we have developed a pipeline for RSO image capture using predictive simulation algorithms applied to low spatial resolution images. This study demonstrates the capability of using predictions via simulation to assist with RSO detection using a low spatial resolution imager. The FAI was used in this study as a sample low spatial resolution camera in orbit with similar characteristics to a commercial grade imager. FAI is currently pointed roughly in an anti-sun configuration which is ideal for RSO imaging and consequently for light curve extraction. We demonstrate this by comparing the pixel locations of RSOs from real images with STK's EOIRtool, and our in-house RSO optical image simulator using (1) predictions with TLEs implemented with anti-sun pointing and (2) ephemeris-based simulations.

With this study, we have demonstrated that by using TLEs and pre-determined orientation modes, we can determine which objects pass by a particular observer. Additionally, this approach allows us to identify some of the RSOs in the vicinity of the observer, making it a valuable tool for future photometric satellite missions.

7.1 Introduction

SSA, which includes predicting, tracking, and identifying RSOs is a very important field to deal with the large increase in the number of RSOs in orbit. Accurate RSO tracking and proximity monitoring are becoming increasingly important, as the growing RSO population causes more frequent conjunction events in which operational spacecraft must maneuver to avoid collisions. More details on the different definitions of SSA and its importance are found in [114].

7.1.1 SSA using low spatial resolution imagers

Low spatial resolution imagers like star trackers have been determined to work well for RSO imaging due to their wide FOVs [116, 124–128] . Additionally, star trackers allow us to use existing hardware widely available on many satellites already in orbit for RSO detection, tracking, and identification in addition to their original purpose for attitude determination [129]. Therefore, by widely adopting preexisting star trackers for SSA algorithms, we greatly enhance our ability to detect and track RSOs.

This in turn allows us to further develop and improve our algorithms without the additional cost of a dedicated high spatial resolution imager.

By continuously monitoring the positions and movements of RSOs relative to the background stars, star trackers can help maintain an up-to-date catalog of objects in orbit. In [125], the opportunity to use star trackers to help SSA was examined through simulation studies. In [128], a camera for both attitude determination and SSA is presented as a concept of operation. With a network of satellites equipped with star trackers, the authors in [130] propose Space Traffic Management (STM) as a means for SSA. Star trackers can contribute to the tracking and identification of objects, allowing space agencies and operators to plan and manage space traffic effectively.

The camera chosen for this study is the FAI onboard the CASSIOPE satellite [86]. Although FAI is not a star tracker, it is a low spatial resolution and wide FOV camera with image quality similar to a star tracker. Additionally, FAI images are publicly available unlike most space cameras. The availability of real images for validation allows us to use the FAI for algorithm testing and development.

7.1.2 Starfield simulation for SSA

One method to assist with SSA activities is to simulate RSO images. These simulations depict an accurate view of what objects are in the FOV of our observer. There are many advantages for using RSO image simulations. First, using simulations validates what RSOs will be in view of our target, which is useful for conjunction warnings and impact prediction. In [131], simulations were implemented to determine the effectiveness of various maneuver guidelines for STM.

Second, simulated images are accessible much more quickly than real data for time-sensitive issues such as collision avoidance, due to delays in downlinking images from orbit, and so simulated images are useful for preliminary analysis. As a result, simulated images are capable of replacing real data temporarily when there are issues with downlink. This allows for a faster response time to potential collisions if they are able to be detected in a simulation environment. Once real images are available, simulated images assist in studying images obtained from real cameras by comparing the real images with the simulated parameters. Most simulators save important information about the target and the observer, along with the simulated starfield images, which might not be available directly in real images. As a result of being in a controlled environment, simulated images allow for controlling the values of different variables to test out different scenarios which is difficult to implement for real images. In [115], simulated RSO images are used to augment real images for use in AI RSO detection.

Third, simulated images allow for trial and error in designing different aspects of a satellite mission. This includes testing different orbital parameters such as inclination or semi-major axis values prior to launch.

Lastly, SSA simulations also enable predictions of future RSO passes, which is essential for tasking satellites to specific targets or planning efficient downlinks to ground stations. Active satellites generate significant amounts of data, and effective downlink scheduling across multiple ground stations is crucial for SSA. The authors in [132] describe a scheduling approach that optimizes downlink to multiple ground stations by considering data rate and efficiency. This kind of optimization is vital for SSA access planning, where the efficient use of satellite and ground station resources is key to maintaining awareness of RSOs in space. Typically, SSA activities are scheduled using optimization techniques, as demonstrated in multiple studies [132–135]. Recently, reinforcement learning approaches have also been applied to sensor scheduling, as seen in [136–138].

In this chapter, we demonstrate a novel method to use an RSO simulator to predict RSO observations.

7.1.3 RSO Simulators in the market

There are many different RSO simulators in the literature such as [26, 27] in addition to STK's EOIR tool which is widely used in the commercial sector. The simulation tools in both [27] and [26] implement models from rendering software which increases the complexity of the simulator. STK-EOIR is a robust tool but it does not easily allow for running multiple iterations in rapid succession due to delays in running the EOIR tool.

In [31], we described a custom, high fidelity simulator that provided images as well as details on the position of the RSOs from any specified observation platform. This simulation technique uses default MATLAB functions alone with no additional packages or interfaces with other software solutions.

After our study in [31], two of CASSIOPE's reaction wheels failed so the satellite is spinning about the sun line and is oriented such that its main solar panel is facing roughly at the sun at all times and FAI is pointing approximately anti-sun [113]. To generate more accurate predictions from FAI, we implemented an anti-sun pointing mode within the simulator and will use this feature in this study. To complement the anti-sun simulated images and to demonstrate the viability of using low spatial resolution imagers for SSA, we will compare real images, STK-EOIR images, and ephemeris-based simulations with the anti-sun simulated images. The metrics we are studying are the pixel locations of stars and RSOs of interest, the angular difference of RSO locations, and the RSO cross-track position difference.

7.1.4 Research Objectives

To demonstrate the accuracy of the prediction simulations for RSO observation prediction, we have performed the following studies:

1. We developed a sequence for mission planning to observe RSOs, including RSO detection predictions, camera planning, and RSO detection with the camera.
2. We demonstrated the above sequence using FAI as the spaceborne observer and SBOIS, first demonstrated in [31] as the main tool.
3. We validated the results by comparing predictions with the real images that FAI collected in February, 2023.

In this study, validation using real FAI images demonstrated that the simulator was capable of predicting when RSOs will be observed by a spacecraft, albeit with some errors in the RSO's exact pixel location. This proves we can use this simulator for future on-orbit tasking. This also demonstrates the validity of using star trackers as RSO observers, as we can predict the position of target RSOs and confirm them with the downlinked images.

7.2 Background

7.2.1 Simulation Architecture

SBOIS is an optical image simulator that generates starfield images as viewed by a specified observer. There are many considerations that were taken into account when developing the simulator which make it an accurate and robust tool for RSO simulations. These include the consideration of various sources of incident light. In addition to the sun, SBOIS considers Earth limb, zodiac, and moon glow which allows for accurate background sky brightness calculations. Another factor that improves the simulator's fidelity is the use of facet models that contain a detailed representation of the RSO's size and shape without implementing complex ray-tracing models. Additionally, the simulator is capable of generating images for both ground and space-based observers, as defined by the user, for a variety of different orientations. For a ground-based observer, one specifies a set of Latitude, Longitude, and Altitude (LLA) in addition to the RA and Dec. For space-based observations, the user can use ephemeris files, containing the spacecraft's measured position and attitude, or TLEs to specify the observer's state. Given that TLEs do not include attitude information, SBOIS has some orientation modes that are activated to estimate the attitude if the observer's position is defined using a TLE. These orientation modes include tracking mode, where the observer is oriented such that a specified target RSO is always in the center of the image, and anti-sun pointing mode, as described in [4].

A file with a collection of variables is saved alongside the images which contains information on the catalog number, position, velocity, and pixel location of the RSOs in the image as well as information on the stars in the image.

7.2.2 Studied Sequences

For this study, we use the FAI onboard the CASSIOPE satellite as our observer. Although FAI was originally developed to observe the aurora, it is now used for viewing RSOs [31, 116]. FAI has a low spatial resolution of 256x256 pixels so FAI images are representative of star tracker images which makes FAI an ideal candidate for this study. Additionally, FAI images and ephemeris data are also publicly available online which allows us to validate our estimated attitude and simulated images with truth data. We simulated FAI in anti-sun pointing mode to generate the predicted images. This configuration is due to CASSIOPE's anti-Sun pointing attitude that was mentioned earlier, resulting from reaction wheel failures. Multiple images were used in this study and the dates and times are provided in Table 7.1.

Table 7.1: List of sequences

Sequence Number	Date	Number of Images in Sequence	Studied Image Timestamp in UTC
1	February 12, 2023	29	04:31:11
2	February 15, 2023	13	21:15:20
3	February 18, 2023	59	19:43:28

7.3 Methodology

This section starts with the explanation of the different dataset that are compared and then explains the list of steps taken to perform the analysis.

7.3.1 Data Sources Used

In this study, four different datasets are compared, namely: predicted images from the RSO simulator, simulated images created using ephemeris data, STK EOIR, and real images.

7.3.1.1 RSO Simulator

The RSO simulator (also known as SBOIS) is a starfield simulator that was initially developed as part of a feasibility study for low spatial resolution imagers. Further details on the RSO simulator are available in [31]. The RSO simulator's functionality has since been expanded to include generating images from ground stations as well as adding the option for tracking mode observations in addition to stare mode. Given that CASSIOPE is currently locked in an approximately anti-sun pointing attitude, the RSO simulator was updated to incorporate an anti-sun pointing mode to match the known CASSIOPE attitude profile. The description of anti-sun pointing is found in [4]. In addition to the anti-sun pointing predictions, the RSO simulator was used to generate simulated images using FAI's ephemeris data which includes its attitude and position. The RSO simulator outputs a simulated image as well as a data file in each timestep containing the needed information for this analysis. To simulate the blurring effect caused by optics, the the RSO location in the simulated image is generated with a Gaussian point spread function. The target catalog number and pixel locations are generated in the data file. A sample image from the RSO simulator is shown in Figure 7.1.

7.3.1.2 STK EOIR

To properly assess the accuracy of the RSO simulator with respect to a commercial tool, we will compare the results with those produced using STK EOIR, which serves as an industry standard for satellite image generation. STK is highly regarded in the space industry as a robust and reliable tool for orbital analysis, with the EOIR tool serving as a very powerful method to visualize what a satellite would see at a specified point in time. Using a custom-designed RSO simulator such as SBOIS is still a

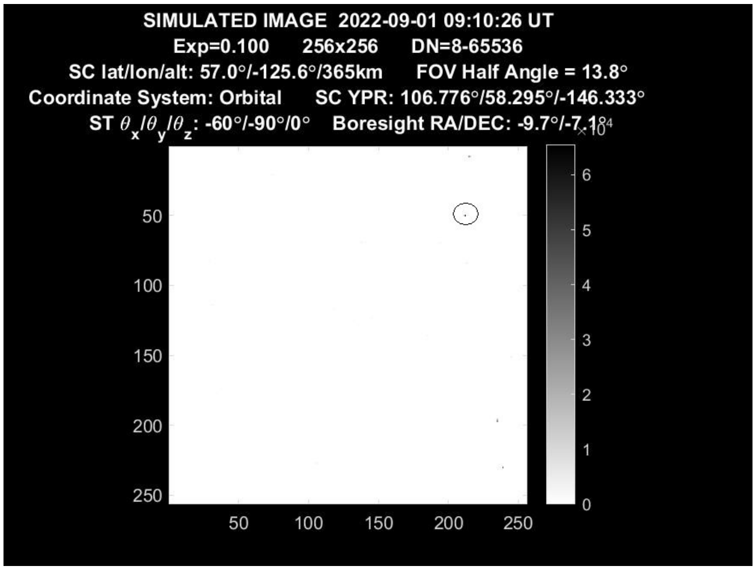


Figure 7.1: Sample image from RSO Simulator with target RSO shown in black circle. Image has been inverted for clarity.

preferred approach for the recurring task of RSO observation predictions and analysis, as it can be optimized in terms of computational speed and modeling fidelity for the task at hand. Nevertheless, as an established industry-standard product, STK EOIR is a valuable tool for validation. A more detailed comparison between an earlier version of the RSO simulator and STK is found in [31]. A sample image from STK-EOIR is shown in Figure 7.2.

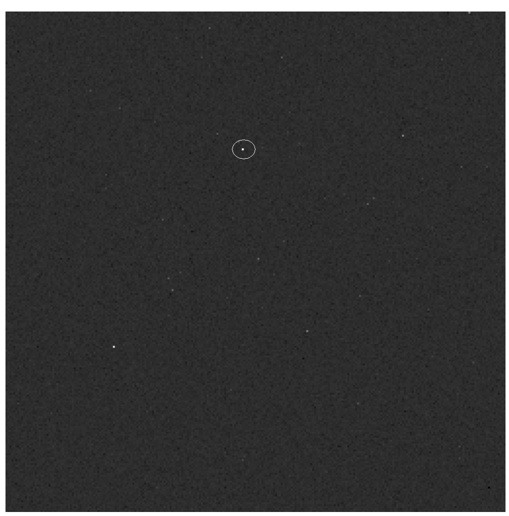


Figure 7.2: Example Image from STK-EOIR with target RSO shown within white circle

7.3.1.3 Real Images

In addition to all the simulation techniques, we also use the real images captured from FAI. The images are uploaded publicly at [119]. Using a sequence of real images, we can manually compare the motion of objects in the images. A sample sequence is in Figure 7.3.

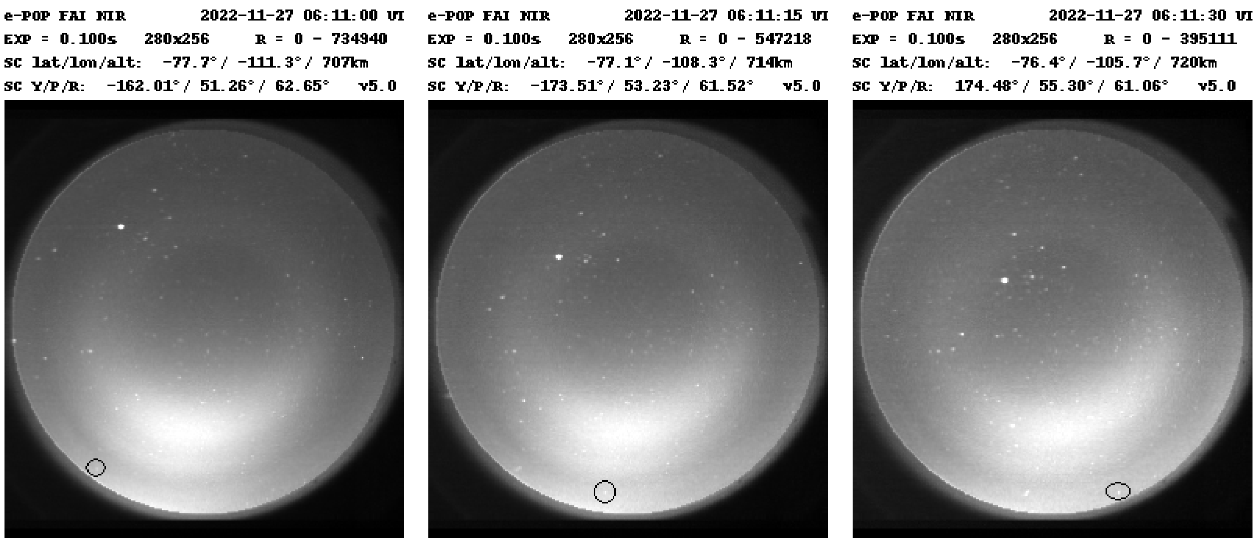


Figure 7.3: Example of sequence of FAI images with target RSO shown within black circle. Unlabeled Image Source: https://epop-data.phys.ucalgary.ca/2022/11/27/FAI/FAI_lv1_png_20221127_060914_061246_6.0.0/

7.3.2 Comparison Approach

The information extracted from the RSO simulator and the real image detections are the pixel locations of the target stars and RSOs at a specified time frame, as well as the RA and Dec of each RSO being tracked. From the RSO simulator, we can also extract the target's catalog number and the star's identity. For the images used in this study, the x and y pixel locations at specified timestamps are used for the comparison while the RA and Dec values are used to calculated the angular difference in RSO locations and the cross track error which is typically used for comparing the positions of satellites [139, 140]. The RA and Dec values are used to determine the angular difference between each RSO's position between the different methods. First, the Cartesian position vector from each method is calculated using Equation 7.1.

$$X = \cos(RA)\cos(Dec) \tag{7.1a}$$

$$Y = \sin(RA)\cos(Dec) \tag{7.1b}$$

$$Z = \sin(Dec) \tag{7.1c}$$

$$\mathbf{V} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{7.1d}$$

Using Equation 7.1, position vectors from each approach for all datasets are calculated and using Equation 7.2 the inverse cosine of the dot product between two of the datasets at a time provides the angular difference in RSO position between the four datasets. To find the difference in cross-track position, which is the difference in position in the plane perpendicular to the line-of-sight to the RSO, we multiply the angular difference in radians with the host-target distance as shown in Equation 7.3. In Equations 7.1 - 7.3, V represents the position vector, θ is the angular difference in radians between two datasets, the subscripts refer to the different datasets being compared, $Error_{CrossTrack}$ is the cross track error in km, and R is the range, or host to target distance, in km.

$$\theta = \cos^{-1}(\mathbf{V_1} \cdot \mathbf{V_2}) \tag{7.2}$$

$$Error_{CrossTrack} = \theta R \tag{7.3}$$

The datasets described earlier were used as follows. First, we started with the RSO simulator to determine potential RSOs in view of FAI. Then, we selected sequences of target RSOs. These were either chosen due to the frequency of the observation or the length of the detection. After determining the sequences to study, we ran STK-EOIR in anti-sun pointing mode for the time window of interest. Concurrently, we provided CASSIOPE's operators with information on the best observation times so that they tasked CASSIOPE to look at the chosen targets. Once images and ephemeris information were captured and downlinked, real images were used for a visual comparison and an updated set of RSO simulations using ephemeris files were compared. Finally, we compared the results from the STK and our in-house simulated images to assess the viability of using low spatial resolution imagers and orientation-specific predictions for SSA. It should be noted that STK-EOIR provides the pixel locations, as well as RA and Dec but requires the user to insert the target manually and to track when the RSO enters and exits the frame to record the needed information.

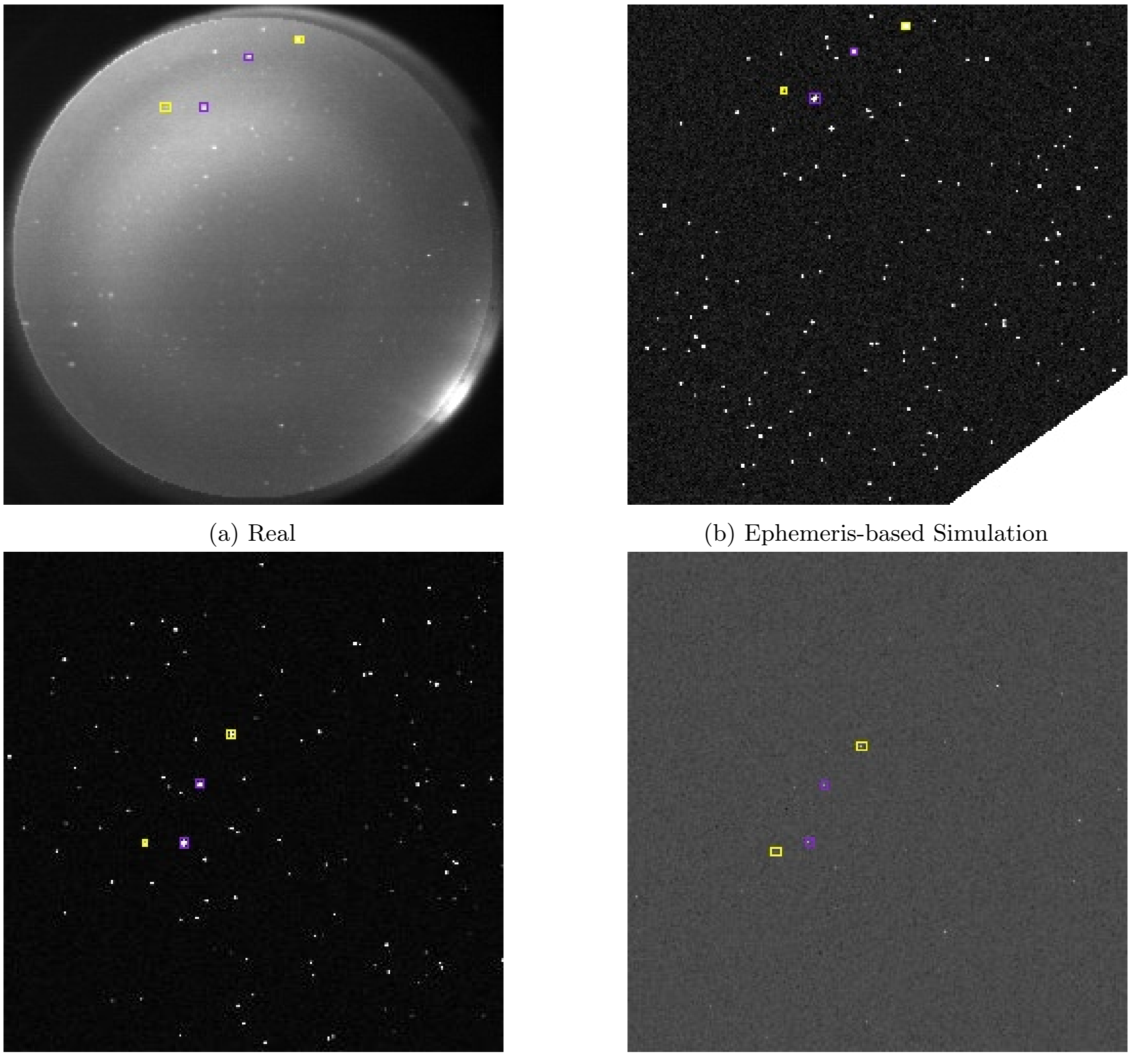
7.4 Results

The real, ephemeris-based simulation, anti-sun pointing simulation, and STK-EOIR images for sequence two are shown in Figure 7.4. In the ephemeris-based simulation shown in Figure 7.4b, the bottom left corner represents the Earth limb. In SBOIS, the Earth limb is modelled by changing the background to white. In future studies SBOIS can be improved to represent an increase in brightness but not necessarily a fully white background. The x and y pixel locations for each target for all sequences are in Table 7.2.

Table 7.2: Pixel locations (x pixel, y pixel) of target stars and RSOs

Image Sequence	Target	Real Image	Ephemeris-based Simulation	Anti-sun Pointing Simulation	STK-EOIR
1	RSO: COSMOS 675	(188,37)	(181,43)	(116,97)	(124,88)
2	RSO: Object Q	(102,53)	(96, 48)	(92,148)	(92,148)
2	RSO: Object L	(126,26)	(116, 24)	(100,119)	(100,119)
2	Star: HR 3982	(151,17)	(141,10)	(116,93)	(119,99)
2	Star: HR 3876	(83,52)	(79,43)	(72,149)	(75,153)
3	RSO: NOAA 6	(103,127)	(104,124)	(53,172)	(55,175)

Figure 7.4 and Table 7.2 show a discrepancy between the anti-sun predictions (both using STK-EOIR and the RSO simulator) and the images from the true attitude profile (real images and ephemeris-based simulations). This indicates that FAI is not entirely pointed in anti-sun configuration. The large pixel differences that are seen between the real and anti-sun simulated images are partly due to attitude control errors where the spin axis will wander by a few degrees with respect to the sun line due to the error in sun vector measurement as well as the imperfections of magnetic torquing. Additional errors



(a) Real (b) Ephemeris-based Simulation (c) Anti-sun Pointing Simulation (d) STK-EOIR

Figure 7.4: FAI images from real and simulated sources for sequence two. Date: February 15, 2023, 21:15:20 UTC. Target stars are in yellow boxes and target RSOs are in purple boxes. The targets are (listed from right to left): Star HR 3982, Object L (catalog number 54692), Object Q (catalog number 54751), Star HR 3876. Source of unlabeled real image: https://epop-data.phys.ucalgary.ca/2023/02/15/FAI/FAI_lv1_png_20230215_211514_211946_6.0.0/

are introduced due to the fact that the anti-sun simulation and STK-EOIR are both assuming a certain spin angle about the sun line that in general will not match the actual spin angle when the images are taken. The largest discrepancy in pixel locations between the real image and anti-sun pointing simulation is 50 pixels in the x direction and 97 pixels in the y direction. The results match each other more closely when comparing the anti-sun images together and the true attitude images with each other. The largest difference between the real images and the ephemeris-based simulations is 10 pixels in the x

Table 7.3: Angular difference (degrees) for each target

Target	Real Image & Ephemeris-based Simulation	Real Image & Anti-sun Pointing Simulation	Real Image & STK-EOIR	Ephemeris-based Simulation & Anti-sun Pointing Simulation	Ephemeris-based Simulation & STK-EOIR	Anti-sun Pointing Simulation & STK-EOIR
COSMOS 675	0.06	0.15	3.16	0.21	3.20	3.03
Object Q	0.31	0.14	0.77	0.43	0.90	0.66
Object L	0.41	0.44	0.99	0.39	0.76	0.56
NOAA 6	0.12	0.15	0.05	0.13	0.16	0.19

direction and nine pixels in the y-direction. The anti-sun and STK-EOIR scenarios display the targets at almost identical locations with the largest difference of eight pixels in the x-direction and nine pixels in the y-direction. This indicates a strong accuracy in the generation of anti-sun pointing simulated images despite the FAI camera not being oriented fully in anti-sun configuration. The difference in the anti-sun pointing and true pointing results in a large shift in target pixel locations where the true attitude shows the targets at the top of the image whereas anti-sun calculations show the targets near the center of the detector.

To further evaluate the simulations, we calculated the difference in look directions between the four data sources for all the target RSOs as shown in Table 7.3.

From Table 7.3, all techniques aside from STK-EOIR result in angular differences less than half of a degree. For the first target only, STK-EOIR has an approximate three degree angular difference to the other techniques. This further supports the previous results in Table 7.2 where the pixel locations in STK-EOIR encountered the largest offset from other methods in the first image sequence. The larger discrepancy is potentially due to TLE propagation errors within STK. The cross-track position difference between the four methods is in Table 7.4.

As shown in Table 7.4, the largest cross-track error is around 37 km. For non-STK comparisons, the error does not exceed five km. This indicates that simulation predictions are a viable technique for determining the locations of target RSOs as, although we are not able to predict the pixel locations

Table 7.4: Cross-Track error (km) for each target

Target	Real Image & Ephemeris-based Simulation	Real Image & Anti-sun Pointing Simulation	Real Image & STK-EOIR	Ephemeris-based Simulation & Anti-sun Pointing Simulation	Ephemeris-based Simulation & STK-EOIR	Anti-sun Pointing Simulation & STK-EOIR
COSMOS 675	0.66	1.78	37.21	1.34	20.83	20.86
Object Q	3.51	1.56	8.73	3.20	6.77	5.33
Object L	4.66	4.97	11.24	2.92	5.7160	4.50
NOAA 6	1.52	1.90	0.60	0.60	0.74	0.89

accurately, the target position in the sky is determined within five km accuracy. With this study we have proven the viability of using TLEs and pre-determined orientation modes to obtain an understanding of what objects pass by a specified observer. Through the example used in this study, we discovered that FAI is not exactly oriented in anti-sun configuration due to inaccuracy in the onboard sun sensor. Even so, we were able to determine some of the stars and RSOs that are in the vicinity of the observer and can use this prediction approach for future satellite missions where the observer is pointed in anti-sun configuration. This study demonstrates the benefits of a simulation environment such as SBOIS and on-orbit star tracker imagery in terms of increased capabilities for SSA activities. While there were differences in the target RSO and star locations, the objects were visible and identifiable in all cases.

7.5 Conclusion

In conclusion, we have demonstrated the feasibility of using low spatial resolution imagers such as star trackers for RSO prediction, simulation, and detection by comparing real FAI images with STK EOIR, and an RSO simulator in anti-sun pointing mode and with ephemeris files. A pixel location comparison indicates a maximum discrepancy of 10 pixels in the x direction and nine pixels in the y direction using the same attitude information. When comparing the cross-track position error in km, the largest difference between the real image and anti-sun simulation is five km while most sequences have errors less than two km. These are promising results that demonstrate the feasibility of using simulations to predict target RSO passes. Additionally, this study provides the initial steps as part of a larger framework to measure and correct for TLE errors. Future work includes adding more pointing orientations such as inertial pointing to allow for predictions of a larger number of RSOs. To further develop the aforementioned framework, it would be crucial to implement an Initial Orbit Determination (IOD) algorithm to further understand the orbits of the RSOs moving around a specified observer.

Chapter 8

Conclusions

8.1 Summary

In conclusion, this dissertation provided contributions to the field of SSA. The contributions in Part I focused on post-mission photometric analysis. Chapters 2 and 3 demonstrated novel ML techniques for LEO RSO light curve classification. SVM, LSTM, and Barlow Twins were explored for light curve classification for this task, with the Barlow Twins network yielding the highest performance. By implementing Barlow Twins, we successfully classified short, two minute light curves with 75% accuracy, and longer five minute light curves with 97% accuracy.

Part II presented software tools for SSA mission planning which were applied to real-world scenarios in collaboration with Magellan Aerospace. Chapter 5 demonstrated a sensitivity analysis study to determine the best camera for achieving SSA mission objectives. Chapters 6 and 7 showcased the use of anti-sun pointing simulations for target prediction, and the results were validated with real images captured from the FAI imager.

8.2 Future Work

For Part I of the thesis regarding the topic of light curves, there are many approaches to enhance the existing research. Firstly, the Barlow Twins network could be tested with different datasets of light curves. This includes developing a tool for light curve extraction to obtain light curves from existing images such as FAI images or images from the RSONAR II mission. The RSONAR II mission served as a technology demonstration of a wide FOV optical camera for SSA. The PCO camera was the main camera that operated as a dataset gathering payload. The IDS was part of a real-time RSO detection and attitude determination system. The light curves studied in this dissertation were obtained from narrow FOV systems, ideal for photometry. However, the PCO, IDS, and FAI cameras are all wide FOV systems which indicate that their images are more challenging to extract light curves from. However, the large FOV allows for the extraction of light curves for multiple RSOs from each sequence of images. Another approach to obtaining light curves for model training is to generate them using the simulator described in Chapters 6 and 7. There are two ways to obtain simulated light curves. The first approach is to generate the simulated images and then extract the light curves and the second approach is to calculate the brightness of the RSOs and simulate the light curves directly without generating images. To use the different datasets for training, there are multiple strategies to be tested such as transfer learning or combined training with all the datasets. Furthermore, adding the capability for online learning, where the Barlow Twins network learns from the light curves as they are collected in real-time, makes the classifier easier to use during missions. This improves the ability of the light curve classifier to generalize to different scenarios which will ensure that satellite operators can reliably use this tool to characterize the motion of RSOs of interest to determine any irregularities.

Regarding the classification task, adding debris light curve as a classification category would allow for a more accurate representation of RSO categories such that any RSO can be classified. In addition to the classification of light curves by object type and stability, other layers of classification can be added to further classify light curves by their shape, size, and surface material. This means the network will be enhanced to perform multi-stage classification where the first network classifies light curves as described in this dissertation with additional networks performing the classification tasks for the other categories. This capability allows for the extraction of as much information as possible with the light curve which, when combined with astrometric analysis from images containing RSO orbital information, allows for more accurate RSO identification. With all these changes, a pipeline is created that starts with capturing images containing RSOs, extracting their light curves, and inputting them one-by-one into the classifier. The classifier will first determine the object type and then the size, shape and material categories.

As for XAI, additional layers in the network can be analyzed by Grad-CAM to allow for a better understanding of how each layer's performance changes after implementing the suggestions above. Other XAI approaches such as SHAP can be utilized to gain a deeper understanding of the impact of different features in the light curve. These approaches can also allow for network pruning to remove layers that do not contribute as much in the new multi-stage classification task.

For Part II, future work includes combining the RSO simulator with the sensitivity analysis tool to emulate the orbit around an in-development spacecraft, combining detectability calculations from the sensitivity analysis with real orbital data from the RSO simulator. This allows for a more accurate study of the true orbital conditions around a spacecraft in development. Other improvements to the detectability analysis tool in Chapter 5 include a more extensive analysis of ideal camera location. The current study used the entry point of target RSOs in the FOV to determine the best camera location with respect to the host's orbit but to further refine the ideal camera location, a study using different positions of the camera on the spacecraft body alongside an analysis of the FOV. An extension of the study performed in Chapter 5 includes a general analysis of common orbital configurations to determine the best location to place a camera depending on the mission type. By having an overall understanding of the best location of an SSA camera, mission operators can choose the orbit more easily. For the study presented in this dissertation, the best camera is the one that detects the most RSOs in one day with a minimum SNR of six but different missions can have different requirements. The detectability analysis tool can be modified to prioritize the number of unique RSOs or the total number of RSOs in the FOV at a particular time. Allowing the analysis tool to be modular will increase the number of applicable mission use cases.

Future improvements to the simulation tool could also involve adding additional pointing modes, such as nadir or inertial pointing. Target-pointing could be implemented in the same approach as anti-sun pointing with the boresight vector aiming at the target RSO as opposed to the inverse sun vector. This is an option to allow the camera to aim at a particular target in each image while in stare mode which is useful in situations where we do not have detailed pointing information but we know the targets that are available in the images. Additionally, a tracking mode would greatly increase the applications of the simulator by expanding the types of studies we can perform with the tool. Further enhancements will also focus on improving the accuracy and speed of RSO predictions.

In summary, combining all the contributions in this thesis results in a comprehensive SSA software solution. This solution starts with sensitivity analysis for camera selection during early mission development, incorporates prediction simulations for mission planning, and ultimately uses light curves to characterize the objects captured from these missions with the developed Barlow Twins network.

Appendices

Appendix A

Distance, Visual Magnitude and SNR Tables

Table A.1: Host to target distance.

RSO Number	Approach Number	Host to Target Distance (km)			
		Entry	Exit	Minimum Rate	Minimum Distance
7	1	247.23	248.46	243.02	238.43
7	2	222.66	217.75	222.66	109.80
7	3	217.78	214.04	170.57	32.73
7	4	232.09	233.32	199.34	158.57
9	1	246.41	236.80	246.41	99.69
18	1	242.23	238.92	204.22	132.59
37	1	244.97	245.23	244.97	241.10
38	1	236.46	243.41	206.46	170.65
38	2	243.98	244.44	243.98	156.92
38	3	235.99	241.38	235.99	198.23
44	1	235.29	245.73	223.56	160.64
44	2	223.33	220.87	223.33	62.57

Table A.1 continued from previous page

RSO Number	Approach Number	Host to Target Distance (km)			
		Entry	Exit	Minimum Rate	Minimum Distance
44	3	248.69	228.64	248.69	53.03
44	4	228.52	232.51	201.05	151.34
53	1	233.91	212.12	233.91	126.64
58	1	244.75	243.34	226.31	158.83
58	2	240.43	242.19	240.43	66.91
70	1	243.10	241.94	243.10	203.78
76	1	207.45	204.91	207.45	40.38
84	1	245.87	241.16	217.91	84.99
84	2	234.40	231.67	234.40	90.59
93	1	245.77	246.86	245.77	223.53
93	2	209.07	227.95	176.45	63.33
94	1	237.99	240.50	237.99	202.78
94	2	226.53	224.39	226.53	142.89

In Table A.1, the first column represents the RSO number out of the 100 propagated orbits. The approach number is a counter for the number of close approaches each target makes. For example, the 7th and 44th RSOs appear within 250 km from the observer a total of four times. While the distances at the entry, exit, and minimum rate points are close to 250 km, the minimum distance is typically much smaller. It is as close as 32 km for RSO number 7's third approach.

Table A.2: Target visual magnitude.

RSO Number	Approach Number	Visual Magnitude			
		Entry	Exit	Minimum Rate	Minimum Distance
7	1	11.14	11.45	11.56	11.59
7	2	9.05	9.35	9.05	9.37
7	3	8.07	8.25	7.69	3.06
7	4	7.62	7.77	7.29	6.52

Table A.2 continued from previous page

RSO Number	Approach Number	Visual Magnitude			
		Entry	Exit	Minimum Rate	Minimum Distance
9	1	8.24	9.59	8.24	7.18
18	1	11.17	8.35	8.16	8.88
37	1	9.67	8.75	9.67	8.98
38	1	7.61	7.79	7.28	6.65
38	2	8.93	9.18	8.93	8.41
38	3	12.04	10.46	12.04	13.76
44	1	9.33	11.22	11.44	12.11
44	2	8.45	9.66	8.45	9.00
44	3	8.17	8.75	8.17	4.48
44	4	7.57	8.08	7.62	6.58
53	1	12.17	8.44	12.17	10.99
58	1	7.61	9.62	9.23	7.11
58	2	7.76	10.21	7.76	4.73
70	1	7.96	7.45	7.96	7.17
76	1	9.12	7.95	9.12	5.91
84	1	7.75	9.28	8.95	5.41
84	2	8.48	10.61	8.48	9.52
93	1	14.91	10.37	14.91	12.39
93	2	9.19	8.06	7.54	6.03
94	1	9.30	7.72	9.30	7.87
94	2	14.98	7.90	14.98	8.98

From Table A.2, most targets are the brightest at the minimum distance point. This does not occur for all approaches though. For example, for the second approach of target RSO number 84, the object appears brightest at the entry point.

Table A.3: Target SNR for FAI.

RSO Number	Approach Number	SNR			
		Entry	Exit	Minimum Rate	Minimum Distance
7	1	0.43	0.32	0.29	0.28
7	2	2.95	2.24	2.95	2.20
7	3	7.29	6.17	10.33	732.92
7	4	11.01	9.61	14.97	30.36
9	1	6.21	1.80	6.21	16.53
18	1	0.42	5.63	6.71	3.47
37	1	1.67	3.91	1.67	3.15
38	1	11.15	9.48	15.13	27.01
38	2	3.31	2.62	3.31	5.35
38	3	0.19	0.81	0.19	0.04
44	1	2.28	0.40	0.33	0.18
44	2	5.12	1.69	5.12	3.09
44	3	6.65	3.88	6.65	199.04
44	4	11.51	7.21	11.05	28.85
53	1	0.17	5.19	0.17	0.50
58	1	11.16	1.75	2.51	17.60
58	2	9.72	1.02	9.72	157.86
70	1	8.05	12.94	8.05	16.71
76	1	2.77	8.15	2.77	53.20
84	1	9.77	2.39	3.24	84.48
84	2	5.01	0.70	5.01	1.92
93	1	0.01	0.88	0.01	0.14
93	2	2.59	7.37	11.90	47.65
94	1	2.35	10.08	2.35	8.76
94	2	0.01	8.51	0.01	3.17

Table A.4: Target SNR for PCO.

RSO Number	Approach Number	SNR			
		Entry	Exit	Minimum Rate	Minimum Distance
7	1	10.14	7.84	7.07	6.78
7	2	74.82	56.47	74.82	25.40
7	3	368.27	311.48	521.85	3346.12
7	4	366.80	320.44	498.96	468.30
9	1	224.75	65.06	224.75	216.07
18	1	9.15	123.17	146.82	38.89
37	1	47.50	106.69	47.50	85.94
38	1	272.75	241.29	385.08	394.70
38	2	71.33	53.33	71.33	69.59
38	3	3.70	15.75	3.70	0.63
44	1	55.58	9.97	8.20	3.23
44	2	183.77	57.86	183.77	22.03
44	3	335.73	196.09	335.73	1307.80
44	4	485.55	332.02	509.31	525.67
53	1	3.39	90.53	3.39	4.76
58	1	563.93	88.34	126.65	889.13
58	2	490.85	51.43	490.85	4583.34
70	1	155.72	247.06	155.72	258.68
76	1	88.63	253.14	88.63	227.81
84	1	493.33	120.51	163.46	1466.44
84	2	238.00	28.62	238.00	34.54
93	1	0.25	16.19	0.25	2.38
93	2	97.69	312.93	505.26	311.34
94	1	57.61	242.60	57.61	197.90
94	2	0.36	244.19	0.36	51.81

Table A.5: Target SNR for IDS.

RSO Number	Approach Number	SNR			
		Entry	Exit	Minimum Rate	Minimum Distance
7	1	0.39	0.30	0.27	0.26
7	2	2.89	2.18	2.89	0.98
7	3	16.13	13.64	22.85	129.15
7	4	14.16	12.37	19.26	18.07
9	1	8.67	2.51	8.67	8.34
18	1	0.35	4.75	5.67	1.50
37	1	1.83	4.12	1.83	3.32
38	1	10.53	9.31	14.86	15.23
38	2	2.75	2.06	2.75	2.69
38	3	0.14	0.61	0.14	0.02
44	1	2.15	0.38	0.32	0.12
44	2	7.09	2.23	7.09	0.85
44	3	14.70	8.59	14.70	50.48
44	4	18.74	12.81	19.66	20.29
53	1	0.13	3.49	0.13	0.18
58	1	24.69	3.87	5.55	38.93
58	2	21.49	2.25	21.49	176.90
70	1	6.01	9.54	6.01	9.98
76	1	3.42	9.77	3.42	8.79
84	1	21.60	5.28	7.16	56.60
84	2	9.19	1.10	9.19	1.33
93	1	0.01	0.62	0.01	0.09
93	2	3.77	12.08	19.50	12.02
94	1	2.22	9.36	2.22	7.64
94	2	0.01	9.42	0.01	2.00

Bibliography

[1] R. Qashoa and R. S. K. Lee, "Classification of Low Earth Orbit (LEO) Resident Space Objects' (RSO) Light Curves Using a Support Vector Machine (SVM) and Long Short-Term Memory (LSTM)," *Sensors 2023, Vol. 23, Page 6539*, vol. 23, no. 14, p. 6539, Jul. 2023, ISSN: 1424-8220. DOI: 10.3390/S23146539. [Online]. Available: https://www.mdpi.com/1424-8220/23/14/6539.

[2] R. Qashoa and R. S. K. Lee, "Low Earth Orbit (LEO) Resident Space Object (RSO) Light Curve Classification using Barlow Twins," *IEEE Transactions on Aerospace and Electronic Systems*, Under Review 2024.

[3] R. Qashoa, P. Harrison, and R. Lee, "Wide Field of View (FOV) Imagers for Co-Orbiting Object Detection," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2024. [Online]. Available: https://amostech.com/TechnicalPapers/2024/Space-Based-Assets/Qashoa.pdf.

[4] R. Qashoa, R. S. K. Lee, and R. Clark, "RSO Simulations with Anti-Sun Pointing Predictions," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2023. [Online]. Available: https://amostech.com/TechnicalPapers/2023/Poster/Qashoa.pdf.

[5] R. Qashoa, R. S. K. Lee, P. Harrison, and M. Driedger, "Demonstration of RSO Simulation, Prediction and Detection Using a Low-Resolution Imager in Anti-Sun Pointing," *Advances in Space Research*, Under Review 2024.

[6] A. Porras-Hermoso, R. Qashoa, R. S. Lee, and J. Cubas, "Accuracy analysis of Space Based Space Debris Tracking for objects in a Low Earth Orbit," *Advances in Space Research*, Under Review 2024.

[7] R. Qashoa, P. Kunalakantha, G. Chianelli, and R. S. K. Suthakar Vithurshanand Lee, "Lessons Learned From Space Situational Awareness Payloads On Stratospheric Balloon," in *Canadian Aeronautics and Space Institute (CASI) ASTRO*, 2024.

[8] R. Qashoa, P. Kunalakantha, G. Chianelli, and R. S. K. Suthakar Vithurshanand Lee, "Lessons Learned From Technology Demonstration Of Space Situational Awareness Satellite," in *Small Satellite Conference*, 2024.

[9] R. Qashoa and R. S. K. Lee, "Comparison of Low Earth Orbit (LEO) Resident Space Object (RSO) Artificial Intelligence (AI) Light Curve Classification Techniques," in *Committee on Space Research (COSPAR) 45th Scientific Assembly*, 2024.

[10] P. Kunalakantha *et al.*, "Resident Space Object (RSO) Tracking in Space-Based, Low Resolution, Non-Constant-Attitude Imagery," *Remote Sensing*, Under Review 2024.

[11] R. Qashoa, V. Suthakar, G. Chianelli, P. Kunalakantha, and R. S. K. Lee, "Technology Demonstration of SSA Mission on Stratospheric Balloon Platform," *Remote Sensing*, vol. 16, no. 5, p. 749, Feb. 2024. DOI: `10.3390/rs16050749`. [Online]. Available: `https://www.mdpi.com/2072-4292/16/5/749`.

[12] R. Qashoa *et al.*, "SPACEDUST-Optical: Wide-FOV Space Situational Awareness from Orbit," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2023. [Online]. Available: `https://amostech.com/TechnicalPapers/2023/Poster/Qashoa2.pdf`.

[13] V. Suthakar, A. A. Sanvido, R. Qashoa, and R. S. K. Lee, "Comparative Analysis of Resident Space Object (RSO) Detection Methods," *Sensors*, vol. 23, no. 24, 2023. DOI: `10.3390/s23249668`.

[14] P. Harrison, R. Qashoa, A. Howarth, R. Clark, and R. S. K. Lee, "Utilizing the CASSIOPE FAI Instrument for Resident Space Object Monitoring," in *Division of Atmospheric and Space Physics (DASP) Workshop*, 2023.

[15] R. Qashoa, M. Vertlib, D. Bédard, and R. S. K. Lee, "Assessing Resident Space Objects (RSO) Stability Using Light Curves," in *Canadian Aeronautics and Space Institute (CASI) ASTRO*, 2022.

[16] R. Qashoa, "Light Curve Analysis Of Resident Space Objects," in *NSERC-CREATE TEPS Conference*, 2022.

[17] ESA Space Debris Office, "Space Environment Statistics," 2024. [Online]. Available: https://sdup.esoc.esa.int/discosweb/statistics/.

[18] N. Johnson, "The Collision of Iridium 33 and Cosmos 2251: The Shape of Things to Come," in *60th International Astronautical Congress*, NASA's Lyndon B. Johnson Space Center, 2009. [Online]. Available: https://ntrs.nasa.gov/api/citations/20100002023/downloads/20100002023.pdf.

[19] J. Gomez and E. DeLetter, "Near-collision between NASA spacecraft, Russian satellite was shockingly close less than 10 meters apart," USA Today, 2024. [Online]. Available: https://www.usatoday.com/story/news/nation/2024/04/23/nasa-timed-spacecraft-russian-satellite-near-collision/73421948007/.

[20] S. Jagtap, "Global Space Situational Awareness (SSA) Market Size & Share Analysis - Industry Research Report - Growth Trends," Coherent Market Insights. [Online]. Available: https://www.coherentmarketinsights.com/industry-reports/global-space-situational-awareness-market.

[21] B. Wallace, L. Scott, and A. Spaans, "The DRDC Ottawa Space Surveillance Observatory," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2007. [Online]. Available: https://amostech.com/TechnicalPapers/2007/Telescopes_Instrumentation/Wallace.pdf.

[22] A. Bloom, J. Wysack, J. Griesbach, and A. Lawitzke, "Space and Ground-Based SDA Sensor Performance Comparisons," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2022. [Online]. Available: https://amostech.com/TechnicalPapers/2022/Poster/Bloom.pdf.

[23] A. Welch, "Challenges of space-based Space Situational Awareness," Raytheon NORSS, 2023. [Online]. Available: https://www.innovationnewsnetwork.com/challenges-of-space-based-space-situational-awareness/34979/.

[24] H. Peng and X. Bai, "Gaussian Processes for improving orbit prediction accuracy," *Acta Astronautica*, vol. 161, pp. 44–56, 2019, ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2019.05.014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0094576518320344.

[25] C. Levit and W. Marshall, "Improved orbit predictions using two-line elements," *Advances in Space Research*, vol. 47, no. 7, pp. 1107–1115, 2011, ISSN: 0273-1177. DOI: https://doi.org/10.1016/j.asr.2010.10.017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0273117710006964.

[26] W. Zhang, Z. Jiang, H. Zhang, and J. Luo, "Optical image simulation system for space surveillance," in *7th International Conference on Image and Graphics, ICIG*, 2013, pp. 721–726, ISBN: 9780769550503. DOI: 10.1109/ICIG.2013.146.

[27] S. P. Hagerty and H. B. Ellis, "A high fidelity approach to data simulation for space situational awareness missions," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2016. [Online]. Available: https://amostech.com/TechnicalPapers/2016/NROC/Hagerty.pdf.

[28] J. Fletcher, I. McQuaid, P. Thomas, J. Sanders, and G. Martin, "Feature-Based Satellite Detection using Convolutional Neural Networks," in *The Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*, 2019. [Online]. Available: https://amostech.com/TechnicalPapers/2019/Machine-Learning-for-SSA-Applications/Fletcher.pdf.

[29] H. Ding, X. Li, and H. Zhao, "An approach for autonomous space object identification based on normalized AMI and illumination invariant MSA," *Acta Astronautica*, vol. 84, pp. 173–181, 2013, ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2012.11.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S009457651200464X.

[30] N. Bartels *et al.*, "Space object identification via polarimetric satellite laser ranging," *Communications Engineering*, vol. 1, no. 1, p. 5, May 2022, ISSN: 2731-3395. DOI: 10.1038/s44172-022-00003-w. [Online]. Available: https://doi.org/10.1038/s44172-022-00003-w.

[31] R. Clark, Y. Fu, S. Dave, and R. Lee, "Simulation of RSO images for Space Situation Awareness (SSA) using parallel processing," *Sensors*, vol. 21, 23 2021, ISSN: 14248220. DOI: 10.3390/s21237868.

[32] A. M. Friedman and C. Frueh, "Observability of Light Curve Inversion for Shape and Feature Determination Exemplified by a Case Analysis," *Journal of the Astronautical Sciences*, vol. 69, 2 2022, ISSN: 21950571. DOI: 10.1007/s40295-021-00293-w.

[33] R. Linares, M. K. Jah, J. L. Crassidis, and C. K. Nebelecky, "Space object shape characterization and tracking using light curve and angles data," *Journal of Guidance, Control, and Dynamics*, vol. 37, 1 2014, ISSN: 15333884. DOI: 10.2514/1.62986.

[34] A. D. Dianetti and J. L. Crassidis, "Space object material determination from polarized light curves," in *AIAA Scitech 2019 Forum*, 2019. DOI: 10.2514/6.2019-0377.

[35] Y. Matsushita, R. Arakawa, Y. Yoshimura, and T. Hanada, "Light Curve Analysis and Attitude Estimation of Space Objects Focusing on Glint," in *The First International Orbital Debris Conference (IOC)*, 2019. [Online]. Available: https://www.hou.usra.edu/meetings/orbitaldebris2019/orbital2019paper/pdf/6091.pdf.

[36] J. Šilha, M. Zigo, T. Hrobár, P. Jevčák, and M. Verešvárska, "Light curves application to space debris characterization and classification," in *8th European Conference on Space Debris*, 2021. [Online]. Available: http://conference.sdo.esoc.esa.int.

[37] N. Cimmino, R. Opromolla, and G. Fasano, "Machine learning-based approach for ballistic coefficient estimation of resident space objects in LEO," *Advances in Space Research*, 2023, ISSN: 18791948. DOI: 10.1016/j.asr.2023.02.007.

[38] R. Clark, Y. Fu, S. Dave, and R. S. K. Lee, "Resident Space Object (RSO) attitude and optical property estimation from space-based light curves," *Advances in Space Research*, vol. 70, 11 2022, ISSN: 18791948. DOI: 10.1016/j.asr.2022.08.068.

[39] A. Ceniceros, D. E. Gaylor, J. Pinon, P. Dao, and R. Rast, "Comparison of BRDF-Predicted and Observed Light Curves of GEO Satellites," in *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2015. [Online]. Available: https://amostech.com/TechnicalPapers/2015/Poster/Ceniceros.pdf.

[40] G. P. Badura, C. R. Valenta, and B. Gunter, "Convolutional Neural Networks for Inference of Space Object Attitude Status," *Journal of the Astronautical Sciences*, vol. 69, pp. 593–626, 2 Apr. 2022. DOI: 10.1007/s40295-022-00309-z.

[41] J. Allworth, L. Windrim, J. Wardman, D. Kucharski, J. Bennett, and M. Bryson, "Development of a high fidelity simulator for generalised photometric based space object classification using machine learning," in *Proceedings of the International Astronautical Congress, IAC*, 2019. [Online]. Available: https://arxiv.org/abs/2004.12270.

[42] B. Liu, L. Yao, and D. Han, "Harnessing ontology and machine learning for RSO classification," *SpringerPlus*, vol. 5, 1 2016, ISSN: 21931801. DOI: 10.1186/s40064-016-3258-2.

[43] R. Furfaro, R. Linares, D. Gaylor, M. Jah, and R. Walls, "Resident Space Object Characterization and Behavior Understanding via Machine Learning and Ontology-based Bayesian Networks," in *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2016. [Online]. Available: https://amostech.com/TechnicalPapers/2016/SSA-Algorithms/Furfaro.pdf.

[44] B. Liu, L. Yao, J. Wu, Z. Hao, and Z. Ding, "From Data Silos to Intelligent Web for RSO Recognition," in *The 6th IIAE International Conference on Intelligent Systems and Image Processing (ICISIP)*, 2018. DOI: 10.12792/icisip2018.023.

[45] N. I. Koshkin *et al.*, "Ukrainian Database And Atlas Of Light Curves Of Artificial Space Objects," *Odessa Astronomical Publications*, vol. 30, 2017, ISSN: 1810-4215. DOI: 10.18524/1810-4215.2017.30.117655.

[46] O. M. Kozhukhov, D. M. Kozhukhov, and V. I. Prysiaznyi, "Ukrainian Optical Sensors for Space Surveillance," in *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2020. [Online]. Available: https://amostech.com/TechnicalPapers/2020/Poster/Kozhukhov.pdf.

[47] U. N. O. for Outer Space Affairs, "Outer Space Objects Index." [Online]. Available: https://www.unoosa.org/oosa/osoindex/index.jspx?lf_id.

[48] S. Jahirabadkar, P. Narsay, S. Pharande, G. Deshpande, and A. Kitture, "Space Objects Classification Techniques: A Survey," in *2020 International Conference on Computational Performance Evaluation, ComPE*, 2020. DOI: 10.1109/ComPE49325.2020.9199996.

[49] P. Dao and K. Knobel, "Test of Neural Network Techniques using Simulated Dual-Band Data of LEO Satellites," in *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2010. [Online]. Available: https://amostech.com/TechnicalPapers/2010/Posters/Dentamaro.pdf.

[50] J. Allworth, L. Windrim, J. Bennett, and M. Bryson, "A transfer learning approach to space debris classification using observational light curve data," *Acta Astronautica*, vol. 181, 2021, ISSN: 00945765. DOI: 10.1016/j.actaastro.2021.01.048.

[51] P. Dao, "Machine Classification and Sub-Classification Pipeline For GEO Light Curves," *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2019. [Online]. Available: `https://amostech.com/TechnicalPapers/2019/Machine-Learning-for-SSA-Applications/Dao.pdf`.

[52] R. Linares, R. Furfaro, and V. Reddy, "Space Objects Classification via Light-Curve Measurements Using Deep Convolutional Neural Networks," *Journal of the Astronautical Sciences*, vol. 67, 3 2020, ISSN: 21950571. DOI: `10.1007/s40295-019-00208-w`.

[53] E. Kerr *et al.*, "Light Curves for GEO Object Characterisation," in *8th European Conference on Space Debris*, vol. 5, 2021. [Online]. Available: `https://conference.sdo.esoc.esa.int/proceedings/sdc8/paper/66/SDC8-paper66.pdf`.

[54] W. D. Bennette, K. Zeliff, and J. Raquepas, "Classification of objects in geosynchronous Earth orbit via light curve analysis," in *IEEE Symposium Series on Computational Intelligence, SSCI*, 2018. DOI: `10.1109/SSCI.2017.8280966`.

[55] K. Balachandran and D. K. Subbarao, "Classification of Resident Space Objects by shape and spin motion using neural networks and photometric light curves," in *8th European Conference on Space Debris*, 2021. [Online]. Available: `http://conference.sdo.esoc.esa.int`.

[56] M. Akin, "Comparison of wavelet transform and FFT methods in the analysis of EEG signals," *Journal of Medical Systems*, vol. 26, 2002, ISSN: 01485598. DOI: `10.1023/A:1015075101937`.

[57] L. Tran, "From Fourier transforms to wavelet analysis: Mathematical concepts and examples," 2006. [Online]. Available: `https://www.whitman.edu/documents/academics/mathematics/trandl.pdf`.

[58] J. M. Blackledge, *Chapter 3 - The 2D DFT, FFT and FIR Filter*, 2005. DOI: `https://doi.org/10.1533/9780857099464.1.50`.

[59] R. N. Toma *et al.*, "Classification Framework of the Bearing Faults of an Induction Motor Using Wavelet Scattering Transform-Based Features," *Sensors*, vol. 22, 22 2022, ISSN: 14248220. DOI: `10.3390/s22228958`.

[60] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Transactions on Signal Processing*, vol. 62, 2014, ISSN: 1053587X. DOI: `10.1109/TSP.2014.2326991`.

[61] Z. Liu, G. Yao, Q. Zhang, J. Zhang, and X. Zeng, "Wavelet Scattering Transform for ECG Beat Classification," *Computational and Mathematical Methods in Medicine*, 2020, ISSN: 17486718. DOI: 10.1155/2020/3215681.

[62] A. I. Sharaf, "Sleep Apnea Detection Using Wavelet Scattering Transformation and Random Forest Classifier," *Entropy*, vol. 25, 2023, ISSN: 10994300. DOI: 10.3390/e25030399.

[63] A. Sepúlveda, F. Castillo, C. Palma, and M. Rodriguez-Fernandez, "Emotion recognition from ECG signals using wavelet scattering and machine learning," *Applied Sciences (Switzerland)*, vol. 11, 2021, ISSN: 20763417. DOI: 10.3390/app11114945.

[64] W. S. Noble, "What is a support vector machine?" *Nature Biotechnology*, vol. 24, 12 2006, ISSN: 10870156. DOI: 10.1038/nbt1206-1565.

[65] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. DOI: 10.1007/BF00994018.

[66] E. Alpaydin, *Introduction to Machine Learning*, 3rd ed. MIT Press, 2014, ISBN: 978-0-262-02818-9. [Online]. Available: https://mitpress.mit.edu/9780262043793/introduction-to-machine-learning/.

[67] L. Palaversa *et al.*, "Exploring the variable sky with linear. III. Classification of periodic light curves," *Astronomical Journal*, vol. 146, 2013, ISSN: 00046256. DOI: 10.1088/0004-6256/146/4/101.

[68] R. C. Staudemeyer and E. R. Morris, "Understanding LSTM – a tutorial into Long Short-Term Memory Recurrent Neural Networks," *arXiv*, Sep. 2019. [Online]. Available: https://arxiv.org/abs/1909.09586v1.

[69] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," *Physica D: Nonlinear Phenomena*, vol. 404, 2020, ISSN: 01672789. DOI: 10.1016/j.physd.2019.132306.

[70] T. D. Pham, "Time–frequency time–space LSTM for robust classification of physiological signals," *Scientific Reports*, vol. 11, 1 2021, ISSN: 20452322. DOI: 10.1038/s41598-021-86432-7.

[71] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM Fully Convolutional Networks for Time Series Classification," *IEEE Access*, vol. 6, 2017, ISSN: 21693536. DOI: 10.1109/ACCESS.2017.2779939.

[72] C. Donoso-Oliva, G. Cabrera-Vives, P. Protopapas, R. Carrasco-Davis, and P. A. Estevez, "The effect of phased recurrent units in the classification of multiple catalogues of astronomical light curves," *Monthly Notices of the Royal Astronomical Society*, vol. 505, 4 2021, ISSN: 13652966. DOI: 10.1093/mnras/stab1598.

[73] J. Wu, X. Y. Chen, H. Zhang, L. D. Xiong, H. Lei, and S. H. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *Journal of Electronic Science and Technology*, vol. 17, 2019, ISSN: 1674862X. DOI: 10.11989/JEST.1674-862X.80904120.

[74] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms Supplementary Materials," in *Advances in Neural Information Processing Systems (NIPS)*, 2012. [Online]. Available: https://papers.nips.cc/paper_files/paper/2012/file/05311655a15b75fab86956663e1819cd-Paper.pdf.

[75] K. Eggensperger *et al.*, "Towards an Empirical Foundation for Assessing Bayesian Optimization of Hyperparameters," in *NIPS workshop on Bayesian Optimization in Theory and Practice*, 2013. [Online]. Available: https://ml.informatik.uni-freiburg.de/wp-content/uploads/papers/13-BayesOpt_EmpiricalFoundation.pdf.

[76] A. H. Victoria and G. Maragatham, "Automatic tuning of hyperparameters using Bayesian optimization," *Evolving Systems*, vol. 12, 1 2021, ISSN: 18686486. DOI: 10.1007/s12530-020-09345-2.

[77] D. Willingham, "Big Data Analysis and Analytics with MATLAB," in *15th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS)*, 2015. [Online]. Available: https://inspirehep.net/files/308b6af00f28b6114fc5262566a45c51.

[78] N. S. Zulkifli and M. Said, "Parallel Processing Problem and Solution-A Case Study on MATLAB Parallel Computing Toolbox Datastores," *Journal of Advanced Computing Research*, vol. 4, pp. 1–3, 2019. DOI: 10.1109/MCOM.2010.5594678. [Online]. Available: http://www.cs.bath.ac.uk/dm4t/w2-presentations/alfonso-.

[79] L. V. D. Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008. [Online]. Available: https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf.

[80] B. J. Erickson and F. Kitamura, "Magician's corner: 9. performance metrics for machine learning models," *Radiology: Artificial Intelligence*, vol. 3, 2021, ISSN: 26386100. DOI: 10.1148/ryai.2021200126.

[81] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," *arXiv*, 2020. [Online]. Available: https://arxiv.org/abs/2008.05756v1.

[82] P. Pawara, E. Okafor, M. Groefsema, S. He, L. R. Schomaker, and M. A. Wiering, "One-vs-One classification for deep neural networks," *Pattern Recognition*, vol. 108, 2020, ISSN: 00313203. DOI: 10.1016/j.patcog.2020.107528.

[83] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes," *Pattern Recognition*, vol. 44, 8 2011, ISSN: 00313203. DOI: 10.1016/j.patcog.2011.01.017.

[84] N. Reimers and I. Gurevych, "Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks," *arXiv*, 2017. [Online]. Available: https://arxiv.org/abs/1707.06799v2.

[85] H. Abbasimehr, M. Shabani, and M. Yousefi, "An optimized model using LSTM network for demand forecasting," *Computers and Industrial Engineering*, vol. 143, 2020, ISSN: 03608352. DOI: 10.1016/j.cie.2020.106435.

[86] L. Cogger *et al.*, "Fast Auroral Imager (FAI) for the e-POP Mission," *Space Science Reviews*, vol. 189, 1-4 2015, ISSN: 15729672. DOI: 10.1007/s11214-014-0107-x.

[87] G. Semmens, "Asteroid-hunting satellite a world first." [Online]. Available: https://www.eurekalert.org/news-releases/518415.

[88] Y. Ruan, M. Hu, C. Yun, and W. Xue, "Preliminary Safety Analysis of Megaconstellations in Low Earth Orbit: Assessing Short-Term and Long-Term Collision Risks," *Applied Sciences*, vol. 14, 2024. DOI: 10.3390/app14072953. [Online]. Available: https://www.mdpi.com/2076-3417/14/7/2953.

[89] NanoAvionics, "How Many Satellites are in Space?" [Online]. Available: https://nanoavionics.com/blog/how-many-satellites-are-in-space/.

[90] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow Twins: Self-Supervised Learning via Redundancy Reduction," in *International conference on machine learning*, 2021. [Online]. Available: `https://proceedings.mlr.press/v139/zbontar21a/zbontar21a.pdf`.

[91] T. A. Hinners, K. Tat, and R. Thorp, "Machine Learning Techniques for Stellar Light Curve Classification," *The Astronomical Journal*, vol. 156, p. 7, 2018, ISSN: 1538-3881. DOI: `10.3847/1538-3881/AAC16D`. [Online]. Available: `https://iopscience.iop.org/article/10.3847/1538-3881/aac16d`.

[92] G. Cabrera-Vives, I. Reyes, F. Förster, P. A. Estévez, and J.-C. Maureira, "Deep-HiTS: Rotation Invariant Convolutional Neural Network for Transient Detection," *The Astrophysical Journal*, vol. 836, p. 97, 2017, ISSN: 0004-637X. DOI: `10.3847/1538-4357/836/1/97`. [Online]. Available: `https://iopscience.iop.org/article/10.3847/1538-4357/836/1/97`.

[93] T. Gabruseva, S. Zlobin, and P. Wang, "Photometric Light Curves Classification with Machine Learning," *Journal of Astronomical Instrumentation*, vol. 9, 2020, ISSN: 22511725. DOI: `10.1142/S2251171720500051`.

[94] A. Mahabal *et al.*, "Deep-learnt classification of light curves," in *IEEE Symposium Series on Computational Intelligence, SSCI*, 2018, pp. 1–8, ISBN: 9781538627259. DOI: `10.1109/SSCI.2017.8280984`.

[95] S. D. Kügler, N. Gianniotis, and K. L. Polsterer, "Featureless classification of light curves," *Monthly Notices of the Royal Astronomical Society*, vol. 451, pp. 3385–3392, 2015, ISSN: 0035-8711. DOI: `10.1093/MNRAS/STV1181`. [Online]. Available: `https://dx.doi.org/10.1093/mnras/stv1181`.

[96] C. Yu *et al.*, "A survey on machine learning based light curve analysis for variable astronomical sources," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, e1425, 2021, ISSN: 1942-4795. DOI: `10.1002/WIDM.1425`. [Online]. Available: `https://onlinelibrary.wiley.com/doi/full/10.1002/widm.1425`.

[97] B. Rodríguez-Feliciano, J. Serna, A. García-Varela, J. Hernández, and J. F. Cabrera-García, "Machine-learning Morphological Classification of TESS Light Curves of T Tauri Stars," *The Astronomical Journal*, vol. 166, p. 189, 2023, ISSN: 1538-3881. DOI: `10.3847/1538-3881/ACF865`. [Online]. Available: `https://iopscience.iop.org/article/10.3847/1538-3881/acf865`.

[98] K. Balachandran and K. Subbarao, "Comparison of Deterministic Methods to Estimate Sidereal Rotation Period from Light Curves," *Journal of Spacecraft and Rockets*, vol. 58, no. 6, pp. 1733–1751, 2021, ISSN: 15336794. DOI: 10.2514/1.A34976. [Online]. Available: https://arc.aiaa.org/doi/10.2514/1.A34976.

[99] B. K. Iwana and S. Uchida, "An empirical survey of data augmentation for time series classification with neural networks," *PLOS ONE*, vol. 16, no. 7, 2021, ISSN: 1932-6203. DOI: 10.1371/journal.pone.0254841. [Online]. Available: http://dx.doi.org/10.1371/journal.pone.0254841.

[100] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond Empirical Risk Minimization," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=r1Ddp1-Rb.

[101] W. Samek, T. Wiegand, and K.-R. Müller, "Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models," 2017. arXiv: 1708.08296. [Online]. Available: https://arxiv.org/abs/1708.08296.

[102] A. Theissler, F. Spinnato, U. Schlegel, and R. Guidotti, "Explainable AI for Time Series Classification: A Review, Taxonomy and Research Directions," *IEEE Access*, vol. 10, pp. 100 700–100 724, 2022, ISSN: 21693536. DOI: 10.1109/ACCESS.2022.3207765.

[103] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, Feb. 2020, ISSN: 1573-1405. DOI: 10.1007/s11263-019-01228-7. [Online]. Available: https://doi.org/10.1007/s11263-019-01228-7.

[104] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 4768–4777, ISBN: 9781510860964. [Online]. Available: https://dl.acm.org/doi/pdf/10.5555/3295222.3295230.

[105] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17, Sydney, NSW, Australia: JMLR.org, 2017, pp. 3319–3328. [Online]. Available: https://dl.acm.org/doi/pdf/10.5555/3305890.3306024.

[106] M. T. Ribeiro, S. Singh, and C. Guestrin, “"Why Should I Trust You?": Explaining the Predictions of Any Classifier,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 1135–1144, ISBN: 9781450342322. DOI: 10.1145/2939672.2939778. [Online]. Available: https://doi.org/10.1145/2939672.2939778.

[107] L. Podaima, “Magellan Aerospace Awarded Redwing Space Domain Awareness Microsatellite Contract From The Government Of Canada,” Magellan Aerospace Corporation, 2023. [Online]. Available: https://magellan.aero/press-release/magellan-aerospace-awarded-redwing-space-domain-awareness-microsatellite-contract-from-the-government-of-canada/.

[108] K. Bernard, “Optical Co-Orbital Measurements in Low Earth Orbit,” Ph.D. dissertation, Carleton University, 2019. DOI: 10.22215/ETD/2019-13521.

[109] P. Kunalakantha, A. V. Baires, S. Dave, R. Clark, G. Chianelli, and R. S. Lee, “Stratospheric Night Sky Imaging Payload for Space Situational Awareness (SSA),” *Sensors 2023, Vol. 23, Page 6595*, vol. 23, no. 14, p. 6595, Jul. 2023, ISSN: 1424-8220. DOI: 10.3390/S23146595. [Online]. Available: https://www.mdpi.com/1424-8220/23/14/6595/htm%20https://www.mdpi.com/1424-8220/23/14/6595.

[110] P. Kervin, D. Hall, M. Bolden, and J. Toth, “Phase Angle: What Is It Good For,” in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2024. [Online]. Available: https://amostech.com/TechnicalPapers/2010/NROC/Kervin.pdf.

[111] W. E. Krag, “Visible Magnitude Of Typical Satellites In Synchronous Orbits,” in *NASA STI/Recon Technical Report N 75*, 1974. [Online]. Available: https://apps.dtic.mil/sti/tr/pdf/AD0785380.pdf.

[112] J. R. Shell, “Optimizing orbital debris monitoring with optical telescopes,” in *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2010. [Online]. Available: https://amostech.com/TechnicalPapers/2010/Systems/Shell.pdf.

[113] “Routine CASSIOPE (Swarm-Echo) Science Operations Come to an End – e-POP on CASSIOPE.” [Online]. Available: https://epop.phys.ucalgary.ca/routine-cassiope-swarm-echo-science-operations-come-to-an-end/.

[114] J. A. Kennewell and B. Vo, "An overview of space situational awareness," in *Proceedings of the 16th International Conference on Information Fusion*, IEEE, 2013, pp. 1029–1036. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6641108?casa_token=qit0hOjgsccAAAAA:gxmAZsgvcysqe-xsZ_1BobAmN6nVaYFJleqFNAUI-cgm97aSxxHnjfmaJRyk_1jtTKV_knG_.

[115] Z. Chen, Y. Yang, A. Bettens, Y. Eun, and X. Wu, "A Simulation-Augmented Benchmarking Framework for Automatic RSO Streak Detection in Single-Frame Space Images," *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2305.00412v1.

[116] S. Clemens, R. Lee, P. Harrison, M. Aerospace, and W. Soh, "Feasibility of Using Commercial Star Trackers for On-Orbit Resident Space Object Detection," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2018. [Online]. Available: https://amostech.com/TechnicalPapers/2018/Space-Based_Assets/Clemens.pdf.

[117] S. Dave, R. Clark, and R. S. Lee, "RSOnet: An Image-Processing Framework for a Dual-Purpose Star Tracker as an Opportunistic Space Surveillance Sensor," *Sensors*, vol. 22, 15 2022, ISSN: 14248220. DOI: 10.3390/s22155688.

[118] M. Cegarra *et al.*, "Attitude Detection of Buccaneer RMM CubeSat through Experimental and Simulated Light Curves in combination with Telemetry Data," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2018. [Online]. Available: https://amostech.com/TechnicalPapers/2018/NROC/Cegarra%20Polo.pdf.

[119] University of Calgary, "e-POP Dataset." [Online]. Available: https://epop-data.phys.ucalgary.ca.

[120] "N2YO USA 40 R/B DEB." [Online]. Available: https://www.n2yo.com/satellite/?s=23159.

[121] "N2YO OneWeb-0647." [Online]. Available: https://www.n2yo.com/satellite/?s=56079.

[122] "eoPortal OneWeb Minisatellite Constellation." [Online]. Available: https://www.eoportal.org/satellite-missions/oneweb#spacecraft.

[123] "N2YO Starlink-6101." [Online]. Available: https://www.n2yo.com/satellite/?s=56115.

[124] S. Palo, G. Stafford, and A. Hoskins, "An Agile Multi-use Nano Star Camera for Constellation Applications," in *27th Annual AIAA/USU Conference on Small Satellites*, 2013. [Online]. Avail-

able: https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=2928&context=smallsat.

[125] D. Spiller *et al.*, "On-orbit recognition of resident space objects by using star trackers," *Acta Astronautica*, vol. 177, pp. 478–496, 2020, ISSN: 0094-5765. DOI: 10.1016/J.ACTAASTRO.2020.08.009.

[126] E. Plotke, P. C. Lai, Y. Chan, R. M. Ewart, K. Miller, and J. Griesbach, "Dual Use Star Tracker and Space Domain Awareness Sensor In-Space Test," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2021. [Online]. Available: https://amostech.com/TechnicalPapers/2021/Poster/Plotke.pdf.

[127] T. Sundberg, S. Lindström, O. Rasmusson, M. Nylund, J. Lyke, and D. Fronterhouse, "The SSA Instrument on SPARC-1-A Star Tracker for Detection and Orbit Determination of Resident Space Objects," in *First Int'l. Orbital Debris Conf*, 2019. [Online]. Available: https://www.hou.usra.edu/meetings/orbitaldebris2019/pdf/6059.pdf.

[128] Space Foundation, "An Innovative, Dual Use Star Tracker for Detecting Dim Objects for Space Situational Awareness (SSA) while Enabling Attitude Determination," 2019. [Online]. Available: https://www.spacefoundation.org/tech_track_papers/an-innovative-dual-use-star-tracker-for-detecting-dim-objects-for-space-situational-awareness-ssa-while-enabling-attitude-determination/.

[129] S. Dave and R. S. K. Lee, "Feasibility of a Virtual Constellation using Small Aperture, Wide Field of View Optical Systems for Space Domain Awareness and Applications," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2022. [Online]. Available: https://amostech.com/TechnicalPapers/2022/Poster/Dave.pdf.

[130] M. Faga, J. Fountain, P. Kennedy, and S. O'keefe, "Space Traffic Management: Assessment of the Feasibility, Expected Effectiveness, and Funding Implications of a Transfer of Space Traffic Management Functions," *National Academy Of Public Administration*, 2020. [Online]. Available: https://s3.us-west-2.amazonaws.com/napa-2021/studies/united-states-department-of-commerce-office-of-space-commerce/NAPA_OSC_Final_Report.pdf.

[131] M. Borowitz, M. Birch, and R. J. Macke, "An investigation into potential collision maneuver guidelines for future Space Traffic Management," in *The Advanced Maui Optical and Space*

Surveillance (AMOS) Technologies Conference, 2021. [Online]. Available: https://amostech.com/TechnicalPapers/2021/Conjunction-RPO/Borowitz.pdf.

[132] S. Spangelo, J. Cutler, K. Gilson, and A. Cohn, "Optimization-based scheduling for the single-satellite, multi-ground station communication problem," *Computers & Operations Research*, vol. 57, pp. 1–16, 2015, ISSN: 0305-0548. DOI: 10.1016/J.COR.2014.11.004.

[133] R. Stottler and A. Li, "Automatic, intelligent commercial SSA sensor scheduling," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2019. [Online]. Available: https://amostech.com/TechnicalPapers/2019/Space-Situational-Awareness/Stottler.pdf.

[134] Y. Chen, G. Tian, J. Guo, and J. Huang, "Task Planning for Multiple-Satellite Space-Situational-Awareness Systems," *Aerospace 2021, Vol. 8, Page 73*, vol. 8, no. 3, p. 73, 2021, ISSN: 2226-4310. DOI: 10.3390/AEROSPACE8030073. [Online]. Available: https://www.mdpi.com/2226-4310/8/3/73.

[135] H. Cai, Y. Yang, S. Gehly, C. He, and M. Jah, "Sensor tasking for search and catalog maintenance of geosynchronous space objects," *Acta Astronautica*, vol. 175, pp. 234–248, 2020, ISSN: 0094-5765. DOI: 10.1016/J.ACTAASTRO.2020.05.063.

[136] T. G. Roberts, P. M. Siew, D. Jang, and R. Linares, "A Deep Reinforcement Learning Application to Space-based Sensor Tasking for Space Situational Awareness," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2021. [Online]. Available: https://amostech.com/TechnicalPapers/2021/Dynamic-Tasking/Roberts-2.pdf.

[137] B. Oakes, D. Richards, J. Barr, and J. Ralph, "Double Deep Q Networks for Sensor Management in Space Situational Awareness," in *25th International Conference on Information Fusion (FUSION)*, 2022, pp. 1–6. DOI: 10.23919/FUSION49751.2022.9841242.

[138] R. Linares and R. Furfaro, "Dynamic Sensor Tasking for Space Situational Awareness via Reinforcement Learning," in *The Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, 2016. [Online]. Available: https://amostech.com/TechnicalPapers/2016/SSA-Algorithms/Linares.pdf.

[139] F. Vonbun, P. Argentiero, and P. Schmid, "Orbit determination accuracies using satellite-to-satellite tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, no. 6, pp. 834–842, 1978. DOI: 10.1109/TAES.1978.308547.

[140] R. Mackenzie and P. Moore, "A geopotential error analysis for a non planar satellite to satellite tracking mission," *Journal of Geodesy*, vol. 71, no. 5, pp. 262–272, Apr. 1997, ISSN: 1432-1394. DOI: 10.1007/s001900050094. [Online]. Available: https://doi.org/10.1007/s001900050094.