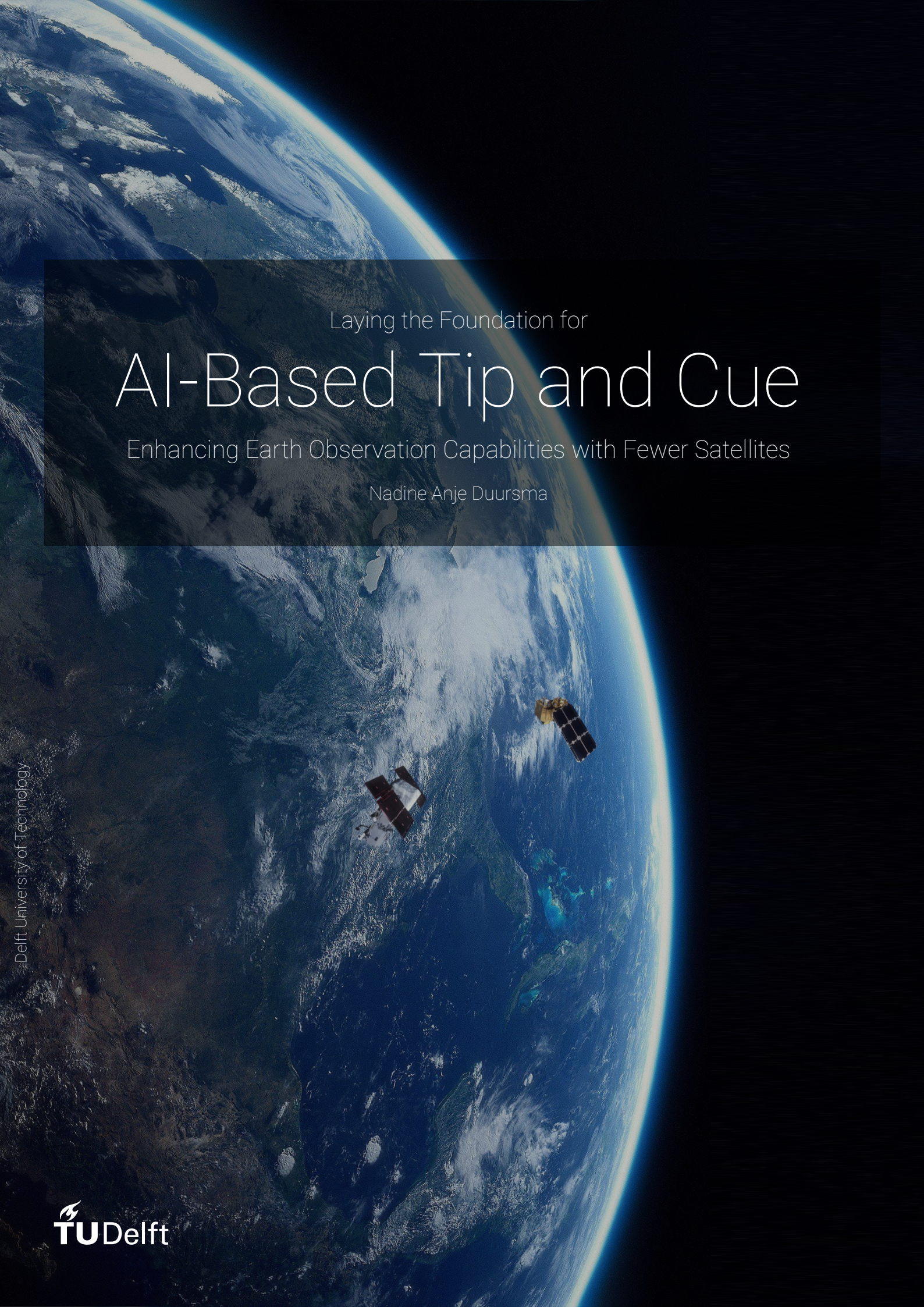


Laying the Foundation for

AI-Based Tip and Cue

Enhancing Earth Observation Capabilities with Fewer Satellites

Nadine Anje Duursma

Delft University of Technology

TU Delft

Laying the Foundation for

AI-Based Tip and Cue

Enhancing Earth Observation Capabilities with Fewer Satellites

by

Nadine Anje Duursma

in partial fulfillment of the requirements for a double degree in

Master of Science
Aerospace Engineering

Master of Science
Robotics

at Delft University of Technology,
to be defended publicly on Wednesday 1 April, 2026 at 12:00.

Student number: 4665236

Thesis committee: Delft University of Technology:
Dr. M. S. Uludag, Space Systems Engineering
Dr. A. Menicucci, Space Systems Engineering
Dr. J. F. P. Kooij, Intelligent Vehicles & Computer Vision
Dr. A. Cervone, Astrodynamics & Space Missions

European Space Agency ϕ-Lab:
Dr. G. Meoni, Onboard Processing & Embedded Systems
Dr. R. Del Prete, Deep Learning & Edge Computing

TU Delft

Preface

Nadine Anje Duursma
Hoofddorp, 17 March 2026

"What research provides in answers, it often gives back in better questions."

That is one of the things I have learned throughout my years at Delft University of Technology. Starting my bachelor's with broad curiosity and limited understanding, and now leaving with a double master's degree, I find myself with more questions about the world than when I began. I mean that in a good way: I now better understand how to identify meaningful problems, how to engage more critically, and how my interests have broadened along the way.

What may be even more important than leaving with more questions is leaving with the skills and tools needed to approach them. That goes far beyond solving equations or writing scripts. Throughout my studies, I have been especially grateful to apply what I learned in different places around the world, from the Arctic to Houston, Princeton, Italy, and Australia. Working with and learning from well over a hundred people from different disciplines and backgrounds taught me to think in systems, to translate between perspectives, and to explore solutions beyond conventional methods.

This also helped me during my thesis work. What began as a new topic proposed by Gabriele gradually developed into something useful for others to build upon. Seeing the framework initiated in this thesis being taken further through the collaboration with the European Space Agency ϕ-lab is something I am quite proud of.

Acknowledgements
I would first like to warmly thank my supervisors Gabriele Meoni and Roberto Del Prete at ESA ϕ-lab, and Julian Kooij and Alessandra Menicucci at Delft University of Technology, for their guidance, involvement, valuable feedback, and steady support throughout this project. I look back very positively on the three months I spent at ESA ESRIN ϕ-lab in Italy during this research.

I would also like to thank the people around me, some nearby in Hoofddorp, others farther away in Andøya or Rome, but always feeling close. Anica, for our weekly swims that helped keep my head sharp. Sanja, for our tea sessions. Roxy, for our shared activities. Gabriele D., for our after-hours conversations. And Annemarie, for always going together. Each of you helped me find balance and, in doing so, helped me better shape this work.

And lastly, to my family. Yuri, my dearest brother, for our endless talks, trust, adventures, confidence, and for inspiring me to think beyond boxes. Papa, for your encouragement, loyalty, and for always believing in me, no matter what I start. And lastly, Mama. For the years we had together, and for everything you taught me. You will always be my brightest star.

Executive Summary

Wildfires spread in hours, ships change course in minutes, and natural disasters evolve faster than most satellites can observe them. Current Earth Observation (EO) systems face a persistent trade-off: wide-area sensors provide frequent coverage but limited detail, while high-resolution satellites provide clear imagery but revisit too slowly. As constellations grow to overcome this trade-off, they also increase orbital congestion and debris risk. This creates a central challenge: how can EO systems respond faster and more precisely without simply launching many more satellites?

This thesis, conducted in collaboration with the European Space Agency (ESA) ϕ-lab, investigates AI-Based Tip and Cue as a strategy. In this concept, a wide field-of-view satellite, the "Tip", first detects a potential event and then autonomously tasks a high-resolution satellite, the "Cue", to acquire targeted follow-up imagery through off-nadir pointing when needed. This combines the coverage of a wide-swath system with the detail of a high-resolution mission.

Despite this potential, AI-Based Tip and Cue remains underexplored. Existing work either relies on ground-based processing, which introduces substantial delay, or includes onboard detection without quantifying how off-nadir viewing geometry affects image quality and onboard detection performance. Prior studies also do not capture the mission-level interaction between orbital spacing, latency, viewing time, and detection success, and no consistent benchmark exists to compare Tip-and-Cue architectures with alternative EO mission designs. In addition, most earlier work treats Tip and Cue as independently operating satellites, even though coordinated formations may be more effective.

This thesis therefore addresses the following research question:

> *How do orbital configuration and off-nadir imaging geometry influence the detection performance and mission feasibility of AI-Based Tip and Cue systems?*

The results show, first, that AI-Based Tip and Cue performance depends jointly on orbital configuration and off-nadir imaging geometry, where detection quality, latency, and viewing time have to be traded off in the design. Second, the results show that, in the choice of mission architecture, Tip-and-Cue formations provide the strongest performance in the tested benchmark. Third, reliable mission evaluation requires representative modelling of off-nadir conditions and target dynamics within an end-to-end evaluation framework.

To investigate this, an end-to-end framework was developed that integrates three components. First, an Orbital Simulation Framework models satellite dynamics, tasking, and pointing, and determines the viewing geometry of each observation. Second, an Off-Nadir Imaging Rendering Pipeline generates imagery under the resulting geometric and radiometric conditions, including perspective changes and sun glint effects. Third, an Onboard Detection Module, based on the transformer-based DEIMv2 model, evaluates detection performance on the generated imagery. In this way, mission geometry, image formation, and onboard detection performance are analysed within a single coupled workflow.

The framework was benchmarked on whale detection. Whales were selected as a demanding demonstration case because they are small, mobile, sparsely distributed, and only briefly visible at the surface, while still requiring high spatial resolution for confirmation. This makes them difficult to monitor at scale and well suited to testing the trade-offs inherent in AI-Based Tip and Cue. To compare configurations consistently, a benchmark was defined around a mission-level metric, successful detection capacity C_{succ}, capturing successful confirmations per satellite per simulated day, supported by latency, viewing time, AP_{50}, precision, and recall.

The framework was validated at multiple levels. The orbital simulation was visualized and checked against expected mission geometries. The off-nadir rendering pipeline was compared with actual Φ-Sat-2 data, showing close agreement in L1C reflectance with median deviations of approximately 1% to 14% depending on spectral band and dataset. The onboard detection model was optimized through staged

hyperparameter tuning, evaluated across four location-based cross-validation folds, and trained for final deployment.

The results show that AI-Based Tip and Cue performance is mainly determined by target accessibility and off-nadir detection reliability. Increasing the maximum Cue off-nadir angle increases the fraction of the Tip window that can be reached and therefore the number of executable Cue tasks, but it also reduces detector performance at larger viewing angles. For the present use case, the favorable off-nadir regime lay between $25°$ and $40°$, depending on ground sampling distance and target visibility. Below that range, not enough of the Tip window could be reached, above it, confirmation performance started to decline. Tip-Cue spacing showed a similar trade-off: shorter delay reduced latency and off-nadir angle, while larger spacing increased follow-up flexibility and viewing time.

The imaging-effect analysis further showed that representative training data is required for realistic Cue evaluation. Models trained on simplified nadir-like imagery overestimated operational performance once tested on representative off-nadir imagery with sun glint. Radiometric realism had a larger effect on transfer performance than geometric realism: omitting radiometric effects reduced AP_{50} from 0.757 to 0.656, whereas omitting geometric effects reduced it to 0.731, and omitting both reduced it further to 0.434 on the test set.

At architecture level, the results showed that a two-satellite Tip-and-Cue system can achieve similar, and in this benchmark slightly outperform, the mission-level performance of a 16-satellite Walker VHR constellation. It achieved a comparable overall successful detection capacity with 87.5% fewer satellites.

The results further show that AI-Based Tip and Cue depends on coordinated detection and follow-up. The 16-satellite Tip-and-Cue formation constellation achieved 20.6 successful confirmations per satellite with a mean latency of 5.0 minutes, whereas the independently operating Tip and Cue satellites dropped to 14.0 at 58.6 minutes. For independent Cue operations, many targets moved out of view before Cue follow-up occurred.

Beyond the whale use case, the off-nadir imaging pipeline also contributes a practical dataset-generation method for data-scarce EO applications. The framework expanded 633 input patches into 13,925 generated images across two classes, each with unique spatial, geometric, and radiometric conditions derived from specific mission geometry. This enables mission-relevant training and evaluation data to be generated already during early mission design, before launch and before large labelled datasets are available.

To extend the work beyond whale monitoring, an applications framework for AI-Based Tip and Cue was introduced to identify where this strategy can create the greatest operational value. 47 EO applications were classified into high, moderate, and limited expected benefit. Wildfire response, illegal activity detection, wildlife monitoring, vessel identification, and rip current alerts ranked among the strongest candidates, because they combine uncertain target location, the need for rapid response, and the need for high spatial detail.

Future work should focus on integrating onboard Tip detection, improving atmospheric modelling and input preprocessing in the rendering pipeline, and extending detection evaluation to larger-scene and multi-class settings. The tasking logic should also be advanced toward target prioritization and image-condition-aware cueing, so that AI-Based Tip and Cue can move from validated feasibility toward mission-specific implementation.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
ADCS	Attitude Determination and Control System
AI	Artificial Intelligence
AIS	Automatic Identification System
AM0	Air Mass 0
AM1	Air Mass 1
AP	Average Precision
AP_{50}	Average Precision at Intersection over Union threshold 0.50
$AP_{50:95}$	Average Precision averaged over Intersection over Union thresholds 0.50 to 0.95
AU	Astronomical Unit
BRDF	Bidirectional Reflectance Distribution Function
BSDF	Bidirectional Scattering Distribution Function
BTDF	Bidirectional Transmittance Distribution Function
CNN	Convolutional Neural Network
COCO	Common Objects in Context
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
DEIMv2	Transformer-based object detection model
DEM	Digital Elevation Model
DN	Digital Number
EO	Earth Observation
ESA	European Space Agency
FN	False Negative
FP	False Positive
GNSS	Global Navigation Satellite System
GSD	Ground Sampling Distance
GPU	Graphics Processing Unit
IoU	Intersection over Union
IUU	Illegal, Unreported and Unregulated
LEO	Low Earth Orbit
LLM	Large Language Model
MPC	Model Predictive Control
ONA	Off-Nadir Angle
OREKIT	Orbital mechanics library used for propagation and geometry
PASEOS	Spacecraft simulation framework used for actor-based mission simulation
PID	Proportional-Integral-Derivative
RAAN	Right Ascension of the Ascending Node
RAM	Random Access Memory
RGB	Red, Green, Blue
RPV	Rahman-Pinty-Verstraete
RTM	Radiative Transfer Model
SAR	Synthetic Aperture Radar
SMARTS	Simple Model of the Atmospheric Radiative Transfer of Sunshine
SPP	Samples Per Pixel
SQ	Sub-question
SSO	Sun-Synchronous Orbit
TLE	Two-Line Element set
TOA	Top of Atmosphere
TP	True Positive
UAV	Unmanned Aerial Vehicle
VHR	Very High Resolution
VRAM	Video Random Access Memory

Symbols

Symbol	Definition	Unit
a	Semi-major axis	m
b_i	Ground-truth bounding box of whale i	px
$\hat{b}_{ij}$	Predicted bounding box j for whale i	px
C_{succ}	Successful detection capacity	detections sat^{-1} day^{-1}
d	Earth–Sun distance	AU
D_i	Set of valid detections for whale i	–
E_0	Solar constant	W m^{-2}
E_{band}	Band-integrated irradiance	W m^{-2}
$\bar{E}_{\text{band}}$	Band-averaged spectral irradiance	$\text{W m}^{-2}\,\text{nm}^{-1}$
$E_{\odot}(\lambda)$	Extraterrestrial spectral solar irradiance	$\text{W m}^{-2}\,\text{nm}^{-1}$
$E_{\text{dir,n}}$	Direct normal irradiance	W m^{-2}
$E_{\text{dif,h}}$	Diffuse horizontal irradiance	W m^{-2}
FN	Number of false negatives	–
FP	Number of false positives	–
h	Altitude	m
L	Mean latency of successful detections	s
$L(\lambda)$	Spectral radiance	$\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}$
L_{band}	Band-integrated radiance	$\text{W m}^{-2}\,\text{sr}^{-1}$
$\bar{L}_{\lambda,\text{band}}$	Band-averaged spectral radiance	$\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}$
m_i	Success indicator for whale event i	–
N_{gt}	Number of ground-truth whales	–
N_{sat}	Number of satellites	–
N_{succ}	Number of successful detections	–
P	Precision	–
Q	Mean matched detection confidence	–
R	Recall	–
$R(\lambda)$	Spectral response function	–
s_{ij}	Confidence score of prediction j for whale i	–
$s_{\min}$	Deployed confidence threshold	–
T	Orbital period	s
T_{sim}	Simulation duration	day
TP	Number of true positives	–
v_i	Viewing time of whale i after successful Cue acquisition	s
V	Mean viewing time of successful detections	s

Symbol	Definition	Unit
α	Full field-of-view angle	°
$\Delta\lambda$	Effective spectral bandwidth	nm
η	Intersection over Union threshold	–
θ	Off-nadir angle	°
θ_S	Solar zenith angle	°
λ	Wavelength	nm
μ	Earth gravitational parameter	$\text{m}^3\,\text{s}^{-2}$
ρ	Reflectance	–
$\rho_{\text{TOA,band}}$	Top-of-atmosphere reflectance in a spectral band	–
τ_i	Latency of whale event i	s
$\tau_{\max}$	Maximum allowed latency	s
φ	Latitude	°

1

Introduction

Earth's natural and human systems are under growing pressure. Climate change is intensifying wildfires, floods, and hurricanes, while illegal fishing, logging, and other human activities continue to degrade ecosystems and natural resources [1], [2], [3], [4], [5]. In these settings, late or low-detail observations limit the ability to assess event severity and respond before impacts grow. [5], [6], [7], [8].

Whale detection is an example of that broader need. Changes in whale abundance signal changes in global ecosystem health [9], [10]. Still, global whale numbers remain highly uncertain [11]. Observing these quantities requires coverage over vast and remote ocean areas, while whales occupy only a very small fraction of the ocean surface and are visible only before submerging again [12], [13]. This makes whale detection from space highly dependent on coverage, timing, and spatial resolution.

This remains a limitation of current Earth observation systems. Wide-swath systems provide area, while very-high-resolution systems provide detail over smaller areas and at lower revisit frequency [14], [15]. Combining both usually requires larger constellations [16]. That creates a paradox: expanding constellations improves monitoring capability, while also increasing cost, operational complexity, and pressure on already congested low Earth and Sun-synchronous orbits [17], [18], [19], [20]. AI-Based Tip and Cue addresses this at the architectural level by separating search from follow-up: a wide field-of-view Tip system detects candidate events, after which a high-resolution Cue system acquires targeted imagery [21], [22].

Still, literature on AI-Based Tip and Cue remains limited, and the studies that exist do not assess it as one connected system. Traditional Tip-and-Cue studies rely on ground processing and human tasking, which limits responsiveness [22], [23], [24]. Studies on onboard AI show that autonomous detection and cueing are feasible, but do not quantify off-nadir image degradation in relation to detection performance [21], [25], [26]. Scheduling studies improve autonomy, but do not couple viewing geometry in their evaluation [27], [28]. It therefore remains unclear when AI-Based Tip and Cue outperforms alternative architectures, and which trade-offs determine that performance.

This thesis addresses that gap by developing an integrated simulation and evaluation framework for AI-Based Tip and Cue. The framework combines orbital simulation, off-nadir imaging rendering, and onboard detection assessment, and introduces a benchmark for comparison with alternative satellite architectures. Whale detection is used as the demonstration case. In this way, AI-Based Tip and Cue can be evaluated against the broader need for observations that can identify events timely with enough detail, to support intervention before impacts escalate.

Following this introduction, Chapter 2 outlines the broader context and introduces AI-Based Tip and Cue. Chapter 3 reviews the existing literature and technical background, followed by Chapter 4, which formulates the research question. Chapter 5 then presents the benchmark, framework, and experimental setup, and Chapter 6 verifies and validates the developed methods. The results are presented in Chapter 7 and interpreted in Chapter 8. Finally, Chapter 9 introduces the applications framework, and Chapter 10 concludes this work.

2

Background

This chapter establishes the context for the research problem and introduces the AI-Based Tip and Cue concept investigated in this work. First, section 2.1 outlines the growing need for timely and high-resolution Earth observation and the limitations of current satellite systems. Building on this, section 2.2 introduces the AI-Based Tip and Cue strategy as a potential approach to overcome these limitations.

2.1. Problem Statement

Over the past decades, the stability of natural and human systems on Earth has increasingly been challenged [1]. Climate change is intensifying extreme weather events and accelerating environmental change [2], leading to more frequent and severe disasters like wildfires [29], floods [30], and hurricanes [31]. At the same time, ecosystems are under growing pressure from human activity, where illegal fishing and logging contribute to biodiversity loss and the degradation of critical habitats [8] [3] [4]. In parallel, geopolitical tensions and increasing economic activity place additional pressure on natural resources and ecosystems worldwide [32]. Together, these developments contribute to environmental degradation, declining ecosystem health, and increasing risks to human societies.

Addressing these challenges requires timely and reliable information about the state of the Earth. Earth observation (EO) satellites provide this capability, where satellite observations enable, for example, early wildfire detection [6], methane leak identification [7], vessel surveillance [8], and disaster response [5] at global scale. However, for many of these societal challenges, detection alone is not sufficient. The observations need to be quick and sharp to accurately classify events, assess the severity, track their evolution, and support mitigation measures before impacts escalate.

This need becomes particularly evident in environmental monitoring applications where events are both small in scale and short in duration. For example, whales are often studied as apex predators that provide insight into the overall health of marine ecosystems, making them important indicators of global ecosystem changes [9], [10]. However, whales frequently submerge and move across vast ocean areas, making them difficult to detect and quantify using conventional observation methods [12]. As a result, despite their ecological importance, the total number of whales in the world's oceans remains highly uncertain [11]. This observational challenge limits the ability to accurately monitor global ecosystem health and to evaluate the effects of environmental degradation on marine biodiversity.

Generally, to achieve both timely, accurate, and large-area monitoring, a satellite system would ideally provide both high spatial resolution and wide swath coverage. However, from a remote sensing design perspective, spatial resolution, swath width, radiometric performance, spectral capabilities, and revisit time cannot all be maximized simultaneously [33]. Wide-swath systems like Sentinel-2 provide large-area coverage and frequent revisit times but moderate spatial resolution [14]. In contrast, very-high-resolution (VHR) systems such as WorldView-3 provide detailed imagery but cover smaller areas and revisit locations less frequently [15]. Achieving both high spatial resolution and rapid revisit therefore typically requires deploying multiple satellites [16].

Therefore, to improve monitoring capabilities, Earth observation systems have expanded rapidly in recent years. Commercial constellations including PlanetScope operate more than 430 satellites [34], the Jilin / Gaofen constellation consists of more than 130 satellites [17], and FireSat aims to deploy dozens of spacecraft dedicated to wildfire detection [35]. Forecasts indicate that the number of EO satellites may triple within the next decade [36]. These constellations aim to increase observation frequency and improve the ability to detect events worldwide.

However, the rapid growth of EO constellations to enhance global monitoring introduces new challenges. Most Earth observation satellites operate in Low Earth Orbit (LEO) or Sun-synchronous orbit (SSO), which are already among the most densely populated orbital regions [18]. Increasing the number of spacecraft in these regions raises concerns about orbital congestion and long-term debris generation [19]. This creates a paradox: improving sustainability and resilience on Earth through enhanced monitoring often relies on launching more satellites, while that same expansion threatens the long-term sustainability of the the environment itself in space [20]. Consequently, simply increasing constellation sizes cannot be considered a sustainable long-term strategy.

This paradox demonstrates the need for novel Earth observation strategies that improve monitoring capabilities without proportionally increasing the number of satellites in orbit. This enables to support the timely detection of environmental change, disasters, and security threats, while maintaining the long-term sustainability of the orbital environment.

2.2. Introduction to AI-Based Tip and Cue

To achieve both high-resolution imaging and high temporal coverage while minimizing the number of satellites, this thesis investigates the AI-Based Tip and Cue strategy.

The Tip and Cue system consists of two satellites with complementary sensor roles. The Tip satellite carries a wide field-of-view sensor and continuously scans a large area at lower spatial resolution. After detecting a feature of interest, it tasks the Cue satellite. The Cue satellite carries a high-resolution sensor and points off-nadir to acquire detailed imagery of the detected target. In this way, large-area monitoring and high-resolution imaging are combined within a two-satellite system, as also illustrated in Figure 2.1.

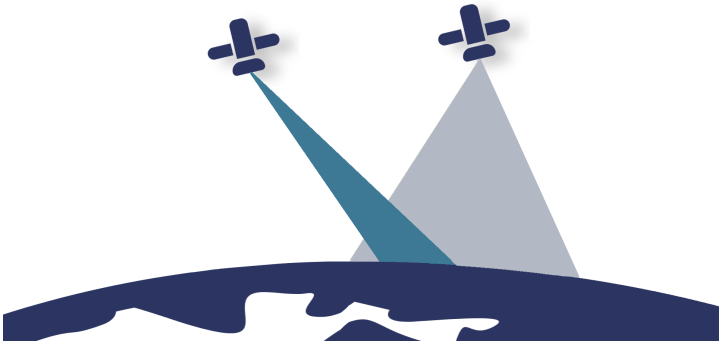


Figure 2.1: Conceptual illustration of the AI-based Tip and Cue formation. The leading Tip satellite scans a wide area with a large field-of-view sensor, while the following Cue satellite performs off-nadir pointing to acquire high-resolution imagery of a detected target.

Traditional Tip and Cue strategies already exist. Typically, a wide-swath satellite detects an event, the data is processed on ground, and a high-resolution satellite from the same or another operator is tasked to acquire follow-up imagery [22], [23]. Because satellites operate in independent orbits and processing occurs on ground, the latency between Tip and Cue observations can reach tens of minutes or even hours in Low Earth Orbit. For dynamic events like wildfires or vessel monitoring, this latency can exceed acceptable limits and illumination conditions may already have changed [24], [37].

An alternative would be to place both sensors on a single satellite. However, this shifts the limitation to onboard processing and pointing constraints. The Cue sensor can only be tasked after the Tip data

have been processed. Ground-based processing introduces downlink and uplink delays, while onboard processing is constrained by mass, power, and computational limitations. For short-lived or fast-moving targets, this may result in missed observations [21].

Therefore, this thesis proposes a formation of two satellites flying in sequence. The Tip satellite precedes the Cue satellite along-track. After detection and onboard processing, the Cue satellite follows at a later time and can point its sensor forward to capture the target. This configuration increases effective viewing time and reduces latency compared to independently operating systems.

A formation-based Tip and Cue system is only feasible with onboard AI. The Tip satellite must detect targets on-board autonomously and transmit compact tasking information to the Cue satellite in real time. Heavy ground-based processing or extensive data transmission is not compatible with the short time window between the satellites.

However, this configuration introduces new trade-offs. Because processing time is limited, the Tip satellite may need to operate on raw or minimally processed data. In addition, the Cue satellite must acquire imagery off-nadir. Off-nadir pointing increases ground sampling distance and introduces geometric distortions such as perspective changes and target elongation. Radiometric effects, including altered illumination geometry, increased atmospheric path length, and sun glint, can further degrade image quality [21].

These effects do not impact all applications equally. Targets that are primarily identified through spatial structure, like vessels or wildlife, may remain detectable under moderate off-nadir angles. In contrast, applications relying on subtle spectral absorption features, such as methane plume detection, are more sensitive to radiometric distortions and atmospheric effects. Larger viewing angles increase atmospheric path length and modify absorption and scattering along the line of sight. Angle-dependent surface reflectance and sun glint can further alter background radiance, complicating the separation between surface reflectance and atmospheric absorption. The suitability of AI-based Tip and Cue therefore depends, besides temporal requirements, also on whether the target is predominantly spatially or spectrally defined.

AI-Based Tip and Cue formations therefore combine advantages: large-area monitoring, high spatial resolution, reduced latency, with challenges related to onboard processing constraints and off-nadir imaging effects. Not all applications benefit equally from this concept. Suitability depends on urgency, spatial scale, and tolerance to off-nadir distortions.

3

Literature Review

This chapter reviews the existing literature relevant to AI-Based Tip and Cue and identifies the gap addressed in this thesis. First, section 3.1 examines prior work on Tip and Cue, covering ground-based implementations, onboard detection, and scheduling and tasking approaches, and shows that the coupling between these elements has not yet been considered. Building on this, section 3.2 introduces the technical background required for the remainder of this work, including off-nadir imaging effects, onboard detection principles, and the whale monitoring application selected as the demonstration case.

3.1. Prior Work on AI-Based Tip and Cue

Prior research on Tip and Cue can broadly be grouped into three categories: traditional ground-based implementations, integration of onboard AI, and optimized scheduling and tasking frameworks. While each of these areas addresses relevant aspects of the concept, they typically focus on isolated components rather than a fully integrated system. This section reviews these categories in more detail, after which the remaining limitations and resulting research gap are summarized.

3.1.1. Traditional Ground-Based Processing Methods

Early Tip and Cue concepts typically rely on ground-processing for detection and tasking. They can combine multiple sensing modalities, where [23] introduced a system integrating Automatic Identification System (AIS), Synthetic Aperture Radar (SAR), and optical imagery for vessel monitoring. Historical AIS data are first analyzed to identify suspicious behavior. Based on this, SAR satellites are tasked to detect non-cooperative vessels, followed by optical satellites acquiring high-resolution imagery for identification.

[22] extended this idea by proposing a 16-satellite constellation consisting of 8 SAR-optical pairs in two orbital planes to enable frequent and high-quality marine monitoring. Their focus lies on generating fused optical-SAR datasets and evaluating them manually, while onboard processing is suggested as future work to reduce latency.

A practical demonstration of Tip and Cue for iceberg monitoring was conducted by [24]. Sentinel-1A imagery was used for tipping, followed by high-resolution ICEYE acquisitions. However, tipping and processing were performed on ground, resulting in significant latency. These approaches therefore demonstrate feasibility but rely on ground-based processing and independent satellites, which limits responsiveness.

3.1.2. Integration of Onboard AI

The introduction of onboard detection enables in-orbit data processing and faster response times. [25] developed a neural network for terrain classification in a multi-tier satellite architecture. The proposed communication protocol involved GEO and multiple LEO satellites to achieve cueing. However, the architecture required several satellites and a ground station, limiting efficiency.

[21] introduced off-nadir pointing in combination with onboard AI for active fire detection. Separate

detection algorithms were developed for detection and segmentation tasks. Their latency analysis showed that a single satellite equipped with dual sensors would not satisfy combined processing and pointing constraints, motivating the need for distributed sensing strategies.

[26] developed the STARCOP model for methane plume detection and demonstrated improved detection performance on higher-resolution Cue imagery. However, while detection performance is evaluated, orbital configuration, latency distributions, and off-nadir distortions are not jointly analyzed. In most cases, detection models are assessed under idealized or static imaging assumptions.

Overall, prior work on integrating onboard AI in traditional Tip and Cue demonstrates that autonomous detection and cueing are technically feasible, but mission geometry and imaging physics are treated separately from detection performance.

3.1.3. Optimized Scheduling and Tasking

Recently, research has focused on automated scheduling and task allocation. These studies investigate how observations can be efficiently assigned under operational constraints.

[27] presented an automated framework that generates tips from external data sources or prior satellite imagery and optimizes the scheduling for Cue satellite tasks. The framework incorporates timing constraints, sensor limitations, and observation utility functions to prioritize targets. Image-based reasoning is integrated into the tasking loop, enabling closed-loop decision-making. While this improves operational autonomy, the satellites are treated as independently operating with different orbits, and imaging distortions due to off-nadir acquisition are not considered.

[28] proposed an autonomous Tip-and-Cue constellation architecture using decentralized reinforcement learning. In this framework, homogeneous satellites carry both wide-field and high-resolution sensors and learn onboard policies to balance scanning for new targets and imaging known ones. The focus lies on collaborative behavior within homogeneous constellations rather than on two-satellite formations with complementary roles.

Although these scheduling approaches advance Tip and Cue autonomy and resource allocation, they do not analyze how off-nadir viewing geometry, latency constraints, and radiometric distortions interact with model detection performance. The physical imaging effects remain largely decoupled from mission-level tasking strategies.

3.1.4. Research Gap

Within existing literature, a separation can be observed between mission-level design and imaging-level modelling.

Studies on onboard detection evaluate model performance but often neglect orbital geometry, latency distributions, and off-nadir distortions. Scheduling studies focus on task allocation and constellation optimization but omit radiometric and geometric imaging effects. Traditional Tip and Cue demonstrations show feasibility but rely on ground-based processing.

To date, no study couples orbital configuration, off-nadir viewing geometry, radiometric modelling, and onboard detection performance within a unified framework.

Specifically:

- The influence of off-nadir viewing angles on onboard detection performance has not been systematically quantified, limiting understanding of how Cue off-nadir geometry affects detection performance.
- The interaction between orbital spacing, latency, and detection feasibility has not been evaluated in an integrated simulation, making it unclear whether timely cueing is practically achievable.
- No benchmark exists to compare AI-Based Tip and Cue systems against alternative satellite configurations, preventing informed mission design and quantitative trade-off analysis.
- No structured overview of potential Tip and Cue applications exists, limiting insight into where the concept can provide the greatest added value.

As long as mission design, off-nadir imaging effects, and onboard detection performance are evaluated independently, their interactions are not captured. However, these components are strongly coupled: orbital design determines the imaging geometry, which directly affects detection quality and, in turn, the overall mission outcome. Consequently, without an integrated end-to-end assessment, the true performance and potential of AI-based Tip and Cue missions remain uncertain.

3.2. Technical Background

To address the identified research gap, both off-nadir imaging effects and onboard detection principles must be considered. In addition, a specific application is required to evaluate these aspects in a realistic context. Based on the application framework presented in Chapter 9 and applications trade-off in Appendix A, whale monitoring is selected as the demonstration case due to its high relevance for ecosystem health, combined with its requirements for high spatial resolution, uncertain target location, and timely response. The following subsections detail off-nadir imaging effects, including geometric and radiometric distortions, and introduce the onboard detection methodologies used in this work.

3.2.1. Off-Nadir Effects

Off-Nadir Angle Definition

The Off-Nadir Angle is defined as the angle between the satellite nadir direction, pointing straight downward to the Earth, and the actual line of sight toward the observed target, as shown in Figure 3.1. A nadir acquisition therefore corresponds to an off-nadir angle of $0°$, while larger values indicate that the sensor is tilted away from the vertical direction [38]. In the AI-based Tip and Cue context, the Cue satellite uses off-nadir pointing to image a target that was previously detected by the Tip satellite, allowing the same area to be revisited without requiring the Cue satellite to fly directly overhead.

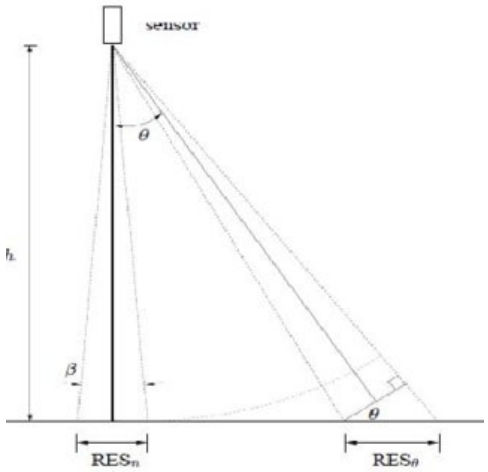


Figure 3.1: Geometry of off-nadir imaging. The Off-Nadir Angle θ is defined as the angle between the satellite nadir direction and the sensor line of sight toward the observed target. Increasing the viewing angle increases the sensor-target distance and enlarges the ground-projected pixel footprint (RES), resulting in a larger ground sampling distance. [39]

Off-nadir imaging introduces a trade-off between responsiveness and image quality. A larger off-nadir angle increases imaging opportunities, reduces revisit constraints, and can reveal side-looking features, shadows, and three-dimensional structure that are not visible in nadir imagery [38], [40]. This additional angular information has proven useful in image classification, object detection, and change detection tasks [41]. At the same time, increasing the off-nadir angle degrades image quality. Geometrically, the sensor-target distance increases, resulting in a larger ground sampling distance and stronger perspective distortion. Radiometrically, the illumination and viewing geometry change, increasing the sensitivity to anisotropic reflection, atmospheric path effects, haze, and sun glint over water surfaces [38].

The suitability of off-nadir imaging therefore depends on the application. In general, low off-nadir angles are preferred when image sharpness and radiometric quality are the main priority, whereas higher off-nadir angles may be acceptable or even advantageous when rapid collection or additional side-view information

is more important [38], [40]. For the whale monitoring use case considered in this thesis, off-nadir imaging is operationally required for rapid cueing, but the associated geometric and radiometric penalties must be quantified to determine the feasible viewing range. The final choice of allowable off-nadir angle must therefore be determined by balancing responsiveness against image degradations for the specific mission objectives.

Geometric Effects

While the Cue satellite operates at off-nadir angles, the image quality will degrade as ground sampling distance increases, making it harder to distinguish whales at lower resolutions. Figure 3.2 shows how increasing ground sampling distance affects image clarity, species identification, and the ability to distinguish individuals in high-density areas [42]. Given that large whale species are sized around 10-20 m [43], and the species in the figure are in the order of 1-3 m, one might consider multiplying the ground sampling distances by this difference factor to maintain a consistent animal-size-to-pixel ratio. However, while the main part of the whale body is below the water surface, only a limited visible surface area of a few metres is typically available for detection, making the effective feature size comparable to the animal examples shown in Figure 3.2.

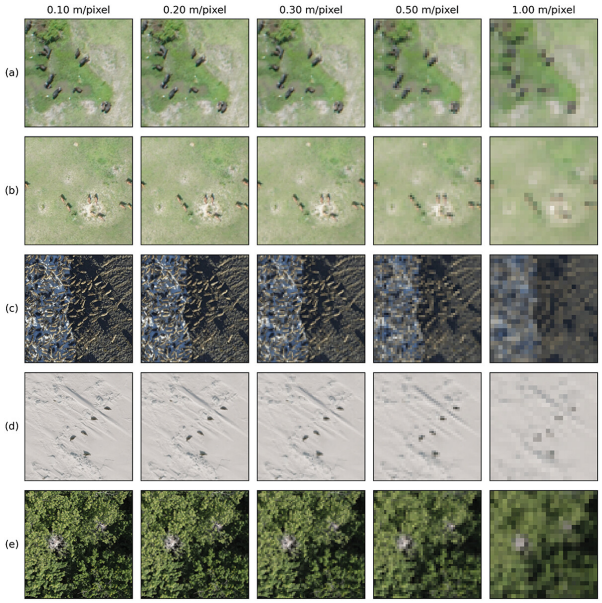


Figure 3.2: Multiple wildlife species under various spatial resolutions, with (a) African buffalo, (b) topi / antelope, (c) caribou / reindeer, (d) harp seal, and (e) nests of great blue heron / bird. The figure illustrates how reduced spatial resolution affects visual interpretability for wildlife monitoring from space [42].

Given the image quality at different spatial resolutions, a ground sampling distance of approximately 50 cm per pixel is expected to be sufficient for detecting the presence of whales. However, finer resolutions are required to distinguish between species, with 10–20 cm per pixel being more suitable for species identification or more precise classification from aerial imagery. The highest commercially available satellite resolution is currently 0.31 m per pixel, as provided by WorldView-3 [15], which has been shown to be sufficient for species detection from space [42].

The increase in ground sampling distance with off-nadir angle is illustrated in Table 3.1 for several Maxar satellites [44]. The table shows that for WorldView-3, a nadir GSD of 0.31 m increases to about 0.50 m at $40°$ off-nadir. This provides a first-order indication of the angular range within which whale detection may remain feasible.

Table 3.1: Ground sampling distance (GSD) for satellites from Maxar Technologies at varying off-nadir angles. Grayed-out boxes indicate GSD values larger than 50 cm, which are considered less favourable for reliable whale detection [44].

Off-Nadir	Ground Sampling Distance (m)			
Angle (deg)	GeoEye-1	WorldView-1	WorldView-2	WorldView-3
0	0.41	0.50	0.47	0.31
5	0.41	0.50	0.47	0.31
10	0.42	0.51	0.48	0.32
15	0.44	0.53	0.49	0.33
20	0.46	0.55	0.52	0.34
25	0.49	0.59	0.55	0.37
30	0.53	0.64	0.60	0.40
35	0.59	0.70	0.67	0.44
40	0.67	0.79	0.76	0.50
45	0.78	0.91	0.90	0.58
50	0.96	1.10	1.12	0.70
55	1.26	1.39	1.51	0.91
60	1.93	1.93	2.51	1.34

While this estimate reflects the geometric loss in spatial resolution, the impact of off-nadir viewing is not limited to a reduction in pixel resolution. Additional perspective distortions from oblique viewing and radiometric disturbances such as glint and haze are not captured in this estimate. Consequently, the actual detection limit cannot be determined from Table 3.1 alone and must be evaluated through the integrated Tip and Cue simulations.

Radiometric Effects

In addition to geometric degradation, off-nadir viewing changes the radiometric appearance of the scene. As the sensor moves away from nadir, the measured radiance becomes more sensitive to shadows, atmospheric scattering, and wavelength-dependent path effects. Over land, this may reveal additional texture and side-structure information, which can improve classification or object detection performance in some cases [41]. Over water, however, one of the most important consequences is the increased risk of sun glint.

Sun glint, as shown in Figure 3.3, occurs when sunlight reflects specularly from the water surface into the sensor, producing bright streaks or saturated regions that can obscure the underlying scene. Figure 3.3 shows an example of this effect. The intensity and location of glint depend on the relative sun position, the sensor viewing geometry, and the local roughness and slope distribution of the water surface [45]. This makes it directly relevant for off-nadir imaging, since changing the viewing angle can either move the sensor into the specular reflection direction and strongly increase glint, or move it away from that direction and reduce it [38], [40]. For the whale monitoring application, this is particularly critical because whales are detected against water with a relatively high reflectance, so glints are expected to be stronger compared to land.

Figure 3.3: Example of sun glint over the ocean surface. Specular reflection of sunlight creates a bright glitter path on the water, which can obscure targets and reduce the interpretability of off-nadir imagery. [46]

Another important radiometric effect is the increase in atmospheric path length with off-nadir angle. Since

the sensor observes the surface through a longer atmospheric column, scattering and absorption effects increase, potentially reducing contrast and introducing haze [38]. These effects are also wavelength dependent, meaning that the spectral content measured by the sensor varies with viewing angle. In conventional remote sensing workflows, atmospheric correction and sun glint normalization are often applied during ground-based processing to compensate for these variations and ensure consistency between scenes. However, these corrections introduce additional processing latency. In the AI-based Tip and Cue context, where rapid response is required, the Cue satellite must operate on imagery that may not yet have undergone full radiometric correction. Therefore, these angular effects need to be represented in the simulation and evaluated in terms of their impact on detection performance.

Bidirectional Reflectance Distribution Function

To model radiometric distortions introduced by off-nadir imaging, like sun glint, the directional reflectance behaviour of the surface must be described. Since the reflected radiance depends on both illumination and viewing geometry, this relationship is commonly represented using the Bidirectional Reflectance Distribution Function (BRDF). The BRDF describes how radiation incident from (θ_i, ϕ_i) is redistributed into direction (θ_r, ϕ_r). This section introduces the BRDF fundamentals required for modelling these radiometric effects in the Tip and Cue simulation framework.

The definition of the Bidirectional Reflectance Distribution Function is given in Equation 3.1 [47].

$$f_r(\theta_i, \phi_i; \theta_r, \phi_r; \lambda) = \frac{dL_r(\theta_i, \phi_i; \theta_r, \phi_r; \lambda)}{L_i(\theta_i, \phi_i; \lambda) \cos\theta_i \, d\omega_i} \tag{3.1}$$

The BRDF defines the ratio between the reflected spectral radiance dL_r in direction (θ_r, ϕ_r) and the incident spectral radiance arriving from direction (θ_i, ϕ_i) within the differential solid angle $d\omega_i$. Here, L_i denotes the incident spectral radiance, while the factor $\cos\theta_i$ accounts for the projected surface area relative to the incoming radiation direction, as illustrated in Figure 3.4 [47].

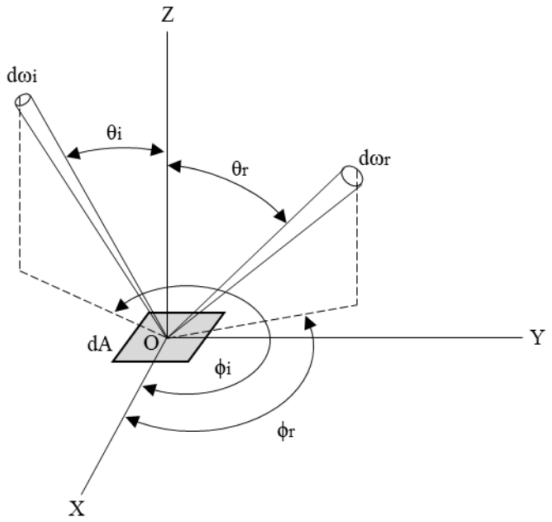


Figure 3.4: Geometry of the Bidirectional Reflectance Distribution Function (BRDF). Radiation incident from direction (θ_i, ϕ_i) within solid angle $d\omega_i$ is reflected into direction (θ_r, ϕ_r) within $d\omega_r$ over surface element dA. [47]

The BRDF depends directly on the incident angles (θ_i, ϕ_i), defined by the solar position, and the outgoing angles (θ_r, ϕ_r), defined by the sensor position. In the AI-based Tip and Cue use-case, it varies with:

- Solar position and intensity, determining (θ_i, ϕ_i) and $L_i(\lambda)$,
- Cue satellite position and viewing direction, defining (θ_r, ϕ_r),
- Sea surface properties like slope and roughness, which modify the local surface normal,
- Wavelength λ, due to spectral dependence of reflectance.

To estimate the reflected radiance that is received by the satellite sensor, Equation 3.1 can be written in integral form as Equation 3.2.

$$L_r(x, y, \theta_r, \phi_r, \lambda) = \int_{\Omega_i} f_r\big(\theta_i(x, y), \phi_i; \theta_r, \phi_r; \lambda\big)\, L_i(\theta_i, \phi_i, \lambda)\, \cos\theta_i(x, y)\, d\omega_i \tag{3.2}$$

Here, Ω_i denotes the hemisphere of incident directions. The local incident angle $\theta_i(x, y)$ depends on the surface normal $\mathbf{n}(x, y)$ defined by the wave geometry. Consequently, L_r varies spatially over the image domain.

Closed-form BRDF parameterisations exist, where f_r can be written explicitly as a function of the angular variables and material parameters. Figure 3.5 illustrates characteristic scattering behaviours represented by commonly used BRDF models. The Lambertian model assumes isotropic diffuse reflection, while microfacet models describe specular reflection based on statistical distributions of surface slopes. Semi-empirical models like the Rahman-Pinty-Verstraete (RPV) formulation introduce additional parameters to represent anisotropic effects, including forward and backward scattering, bell- or bowl-shaped angular responses, and hotspot behaviour, that must be estimated from multi-angular observations or laboratory measurements. [48]

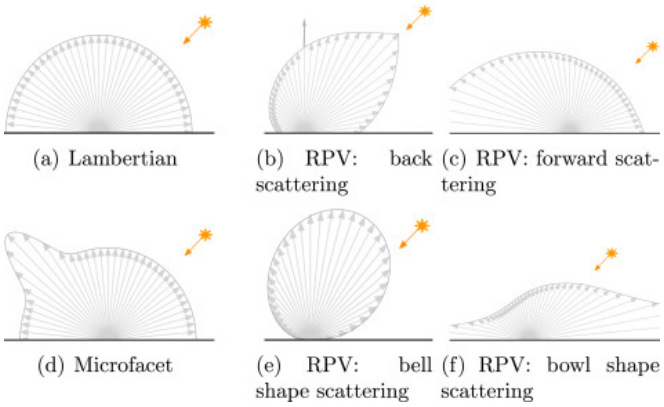


Figure 3.5: Characteristic scattering patterns of common BRDF models. (a) Lambertian diffuse reflection, (b-c) RPV model showing backward and forward scattering behaviour, (d) microfacet model with specular peak, and (e-f) RPV bell-shaped and bowl-shaped angular responses. The orange arrow indicates the illumination direction. [48]

The simplest case, the Lambertian model, given in Equation 3.3, assumes perfectly diffuse reflection and is independent of θ_i, θ_r, and ϕ [45].

$$f_r(\lambda) = \frac{\rho_d(\lambda)}{\pi} \tag{3.3}$$

Here, $\rho_d(\lambda)$ represents the diffuse reflectance. This model does not describe directional effects like specular reflection or sun glint, therefore it is unsuitable for the AI-based Tip and Cue rendering pipeline.

In the case of water, microfacet BRDF models are commonly used to account for directional surface reflection. In the geometric optics approximation, the surface is represented as a distribution of specular microfacets. The surface-reflection component can be written as Equation 3.4 [49]:

$$S(\hat{\omega}_i, \hat{\omega}_s) = D(\hat{\omega}_h)\, F(\theta_d, \tilde{n}(\lambda))\, G(\hat{\omega}_i, \hat{\omega}_s)\, \sigma(\theta_i, \theta_s) \tag{3.4}$$

Here, D is the microfacet normal distribution, F the Fresnel term (depending on the refractive index $\tilde{n}(\lambda)$), G the geometric attenuation (shadowing and masking), and σ the cross-section conversion factor. For isotropic surfaces, D depends only on the half-angle θ_h.

A common closed-form choice for D is the Beckmann distribution as given in Equation 3.5 [49].

$$D_b(\theta_h) = \frac{1}{\pi m^2 \cos^4 \theta_h} \exp\left[-\left(\frac{\tan \theta_h}{m}\right)^2\right] \tag{3.5}$$

where m is a surface roughness parameter. Combining Equation 3.4 with a diffuse term from Equation 3.3 gives a closed-form microfacet BRDF Equation 3.6

$$f_r(\hat{\omega}_i, \hat{\omega}_s, \lambda) = \rho_s \, D(\theta_h) \, F(\theta_d, \tilde{n}(\lambda)) \, G(\hat{\omega}_i, \hat{\omega}_s) \, \sigma(\theta_i, \theta_s) + \frac{\rho_d(\lambda)}{\pi} \tag{3.6}$$

Here, the incident and reflected zenith angles $\theta_i(x, y)$ and $\theta_r(x, y)$ depend on the local surface normal $\mathbf{n}(x, y)$ defined by the wave geometry. As a result, dL_r varies spatially across the image domain.

Although closed-form analytical BRDF expressions like Equation 3.3 and Equation 3.6 exist and are physically motivated, it should be noted that they remain model approximations. Even with more advanced models [45] [49] [50], experimental studies have shown that standard analytical models do not consistently reproduce measured BRDF behaviour across wavelengths and viewing angles, especially near the specular peak and at glossy surfaces, indicating that the assumed microfacet distribution do not fully captures the underlying physics. [51]. This introduces the need for more advanced BRDF modelling methods, which will be explored in subsection 5.3.1.

3.2.2. Onboard Detection

Considerations and Constraints

Onboard detection refers to performing automated image analysis directly on the satellite, instead of transmitting the full imagery to the ground for processing. In the context of AI-based Tip and Cue, this capability is required for reducing response time.

Compared to conventional ground-based processing, onboard detection introduces several additional constraints. Satellite computing platforms have limited computational resources, memory, and power availability [52]. Detection models must therefore be lightweight, minimizing parameter count, memory footprint, and inference latency while still maintaining sufficient detection accuracy [53].

At the same time, detection must be performed within a limited time window, since the Cue satellite may only have a short opportunity to trigger a follow-up observation before the target moves out of the sensor field of view. This constrains deployment to low-level image products, requiring the detection model to be robust to variations in viewing geometry, illumination conditions, and radiometric distortions [54].

Deep Learning Methodologies

Deep learning approaches for satellite-based whale monitoring can be grouped into classification, detection, and segmentation methods. Classification determines whether an image or image patch contains a whale. Detection additionally estimates the target location through bounding boxes or point coordinates. Segmentation predicts the target extent at pixel level. For AI-based Tip and Cue, detection is the most relevant category, since cueing requires providing a location that can be used for follow-up observations. Segmentation may still be useful for higher-level interpretation tasks, like analyzing body shape or estimating visible body extent, but it is typically more computationally demanding.

These tasks are implemented using deep neural network architectures that learn visual features directly from image data. The most commonly used architectures are Convolutional Neural Networks (CNNs), Vision Transformers (ViTs), and hybrid models that combine both approaches.

Convolutional Neural Networks A Convolutional Neural Network (CNN) processes an image by applying convolutional filters to local image regions, followed by nonlinear activation and spatial downsampling. By stacking these operations over multiple layers, the network learns hierarchical feature representations, where early layers respond to simple patterns such as edges and textures, and deeper layers respond to more complex structures [55]. A schematic example is shown in Figure 3.6.

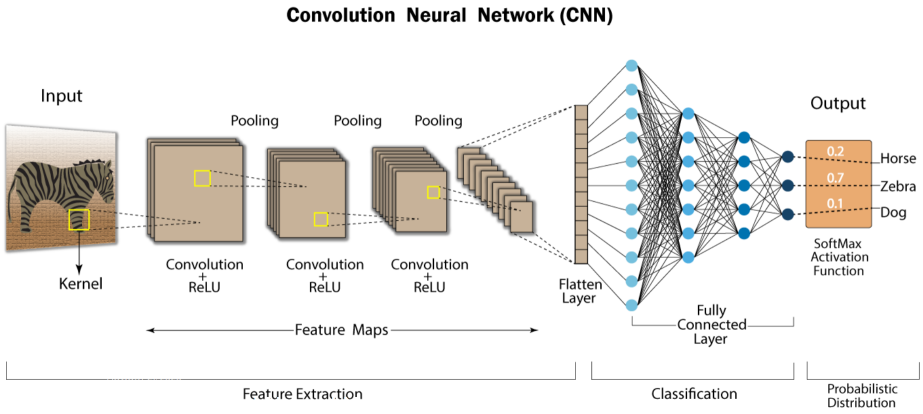


Figure 3.6: Example of a Convolutional Neural Network (CNN) for image classification. The image is processed through successive convolution, activation, and pooling stages to extract hierarchical feature maps, which are then passed to fully connected layers for final class prediction [56].

In the first stage, convolutional feature extraction, the input image is processed using learnable filters, also called kernels. Each kernel slides across the image and computes a weighted sum over a local neighborhood of pixels. This produces a feature map that highlights where a specific visual pattern occurs. During training, the kernel weights are optimized so that different filters respond to useful local structures, for example edges, corners, textures, or small shapes.

In the second stage, nonlinear activation, the output of the convolution is passed through an activation function, most commonly the Rectified Linear Unit (ReLU). ReLU replaces each value x by 0 if $x < 0$ and leaves it unchanged otherwise. This introduces nonlinearity, allowing the network to model more complex visual relationships than a purely linear system.

In the third stage, spatial downsampling, the spatial size of the feature maps is reduced, typically using pooling or strided convolutions. Pooling summarizes values within a small region, for example by taking the maximum value in a 2×2 window, while strided convolutions reduce resolution by applying the filter at spaced intervals. These operations reduce computational cost and enlarge the effective receptive field, so that deeper layers incorporate information from larger regions of the input image.

By stacking convolution, activation, and downsampling layers, CNNs progressively transform local pixel information into higher-level feature maps. These feature maps can then be passed to a task-specific prediction head, for example to produce class scores in classification or object locations in detection. [55], [57], [58]

Vision Transformer Based Models A Vision Transformer (ViT) processes an image in four stages: patch embedding, positional embedding and class token insertion, transformer encoder processing, and classification [59]. Its architecture is shown in Figure 3.7.

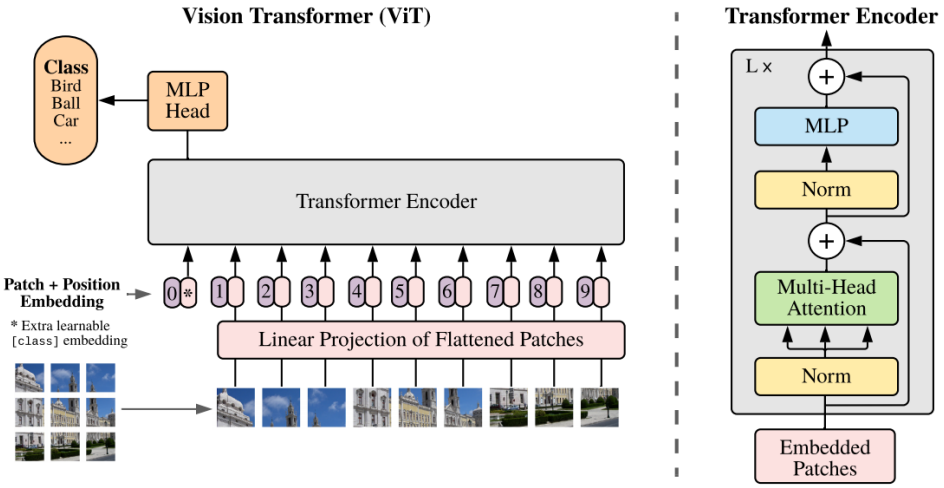


Figure 3.7: Architecture of the Vision Transformer (ViT). The input image is split into fixed-size patches, which are flattened, linearly projected, and combined with positional embeddings. A learnable class token is prepended to the token sequence, which is then processed by transformer encoder blocks consisting of multi-head self-attention and multilayer perceptron layers. The final representation of the class token is used for image classification [59].

In the first stage, patch embedding, the input image is divided into fixed-size patches, for example 16×16 pixels. If the input image has spatial dimensions $H \times W$ and patch size $P \times P$, this results in $N = HW/P^2$ patches. Each patch is flattened into a vector and linearly projected into a feature space of dimension D. These projected patch representations are referred to as patch embeddings and form the input tokens of the model.

In the second stage, positional embedding and class token insertion, the sequence of patch embeddings is augmented in two ways. First, a learnable positional embedding is added to each patch token to retain information about its spatial location in the image. Second, a learnable classification token is prepended to the token sequence. This additional token is intended to aggregate information from all image patches during transformer processing.

In the third stage, the full token sequence is processed by a stack of standard transformer encoder blocks. Each block consists of a multi-head self-attention module followed by a multilayer perceptron (MLP) block. Layer normalization is applied before each block, and residual connections are applied after each block. In the self-attention module, each token is linearly projected to query, key, and value vectors. Attention weights are then computed from the scaled dot products between queries and keys, normalized with a softmax function, and used to form weighted combinations of the value vectors. This allows each token to incorporate information from all other tokens in the sequence. The MLP block then applies two linear transformations with a nonlinear activation function to each token independently.

In the final stage, classification, the output corresponding to the classification token after the last encoder block is used as the image representation. This representation is passed to a task-specific prediction head, typically a linear layer for classification, to produce the final class probabilities. [59], [60], [61], [62]

CNN vs Vision Transformer Comparison Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) differ in how spatial relationships in images are modeled. CNNs extract features locally by applying convolutional filters to small neighborhoods of pixels, while larger spatial context is incorporated gradually as the receptive field increases in deeper layers. Vision Transformers instead divide an image into patches and model interactions between these patches using self-attention, enabling relationships between distant image regions to be captured directly [63].

These architectural differences lead to distinct properties. CNNs rely on local convolution operations with shared weights, which introduces a strong spatial inductive bias and typically results in good data efficiency, particularly when training data are limited [64]. However, their focus on local feature extraction

can make it more difficult to capture long-range spatial dependencies without deeper architectures or additional mechanisms [63]. In contrast, Vision Transformers can model global spatial relationships through the self-attention mechanism, but this flexibility often requires larger datasets or pre-training to achieve competitive performance [63].

For Earth observation imagery, these characteristics influence how targets are detected under varying imaging conditions. CNN-based models are well suited to identifying objects defined by local spatial structures, like edges or textures, which is beneficial for detecting visually distinct targets. Vision Transformers can incorporate broader spatial context, which may improve robustness in scenes where large-scale background patterns influence the interpretation of local features. However, both architectures remain sensitive to radiometric disturbances such as glare or strong reflections that alter local contrast in the image.

Hybrid Architectures Hybrid CNN-transformer architectures combine convolutional layers with transformer blocks within a single model. In typical designs, convolutional layers first extract feature maps from the input image, which are then converted into tokens and processed by transformer layers [65]. Other architectures integrate both mechanisms in parallel or hierarchical structures, allowing local feature extraction and global context modeling to be combined within the same network [66].

Detection Frameworks

Building on the architectural principles introduced in subsection 3.2.2, practical onboard detection pipelines typically consist of two architectural components: a backbone that extracts visual features, like edges, textures, and object shapes, from the image, and a task-specific head that performs detection or segmentation based on those features [67].

In most cases, the backbone is first pretrained on a large generic dataset, after which the pretrained weights are used as initialization and updated using a smaller labelled whale dataset [68]. The backbone may optionally be fine-tuned to adapt the learned representations to the specific characteristics of whale imagery. This transfer-learning strategy is especially relevant for whale detection, where labelled satellite imagery remains limited [69].

CNN Backbones CNN-based backbones remain the most widely used feature extractors in object detection pipelines because they are computationally efficient and comparatively data-efficient, as discussed in subsection 3.2.2. Common CNN backbones include ResNet [70], Inception-v3 [71], DenseNet [72], and EfficientNet [73]. These architectures differ mainly in how they balance representational capacity and computational cost. ResNet introduces residual connections that enable stable training of deep networks [70], Inception-v3 captures multi-scale information through parallel convolution branches [71], DenseNet promotes feature reuse through dense connectivity between layers [72], and EfficientNet improves parameter efficiency through compound scaling of network depth, width, and input resolution [73]. In detection pipelines, these networks are typically used as pretrained feature extractors.

Transformer Backbones Transformer-based backbones can model spatial relationships through self-attention mechanisms. Common Vision Transformer backbones include ViT [59], DeiT [74], DeepViT [75], and the Swin Transformer [76]. These architectures differ mainly in how they address the computational and data requirements of transformer-based vision models. ViT divides an image into fixed-size patches that are processed through global self-attention to capture long-range dependencies [59]. DeiT improves data efficiency by introducing a knowledge distillation strategy that allows transformers to be trained effectively on smaller datasets [74]. DeepViT enhances training stability in deeper transformer networks by increasing diversity in attention maps across layers [75]. The Swin Transformer introduces shifted local attention windows and a hierarchical architecture, improving computational efficiency and making it particularly suitable for dense prediction tasks [76]. [77].

Detection Architectures After feature extraction by the backbone, object detection architectures use the generated feature maps to localize objects within the image. These architectures combine the backbone with additional components that predict object locations and classes.

For CNN, main detector families are commonly distinguished. Two-stage detectors, like Faster R-CNN [78], first identifies candidate regions and subsequently refine their classification and localization. This approach often achieves high detection accuracy, particularly for small or difficult targets, but requires multiple processing stages and therefore increases inference time. One-stage detectors, like YOLO [79], [80], instead perform object localization and classification in one single forward pass. This significantly reduces inference time, making one-stage detectors attractive for onboard applications where latency, memory use, and power consumption are limited. For AI-based Tip and Cue, this speed-accuracy trade-off is critical because the detector must provide reliable target locations within a short response window.

Transformer-based detectors use transformer modules directly for object detection, not only as feature extractors. Detection Transformer (DETR) [81] is the main example. It combines a pretrained CNN backbone, typically ResNet. The CNN backbone first extracts feature maps from the image, which are then processed by the transformer to model relationships between image regions. Subsequent variants including DINO [82] and RT-DETR [83] improve convergence and detection accuracy through denoising training strategies, improved object query design, and architectural changes that reduce inference latency. While transformer-based detectors simplify the detection pipeline and can achieve strong performance, their higher computational cost can limit their applicability in resource-constrained onboard environments.

Segmentation Architectures When more detailed spatial information is required, segmentation architectures can be used to delineate the visible extent of whales within the image. U-Net [84] uses an encoder-decoder architecture with skip connections to preserve spatial detail during upsampling, while U-Net++ [85] extends this design through nested skip pathways that improve multi-scale feature fusion. Mask R-CNN [86] combines object detection with instance segmentation by predicting a pixel-level mask for each detected object. These models provide more detailed spatial output than bounding-box detectors, but their higher computational cost makes them less suitable for the primary tasking step in a low-latency onboard pipeline. They are therefore more relevant for detailed downstream analysis or ground-based processing.

Evaluation Metrics

Detection performance is evaluated using bounding-box based object detection metrics. Overlap between predicted and ground-truth boxes is quantified using the Intersection over Union (IoU), defined in Equation 3.7. The IoU represents the ratio between the area of intersection and the area of union of the predicted and ground-truth bounding boxes, where A_{pred} and A_{gt} represent the respective box areas [87].

$$\text{IoU} = \frac{A_{\text{pred}} \cap A_{\text{gt}}}{A_{\text{pred}} \cup A_{\text{gt}}} \tag{3.7}$$

For each image, detections are matched to ground-truth annotations using an IoU threshold, where box overlap above a certain IoU is considered as positive, and below, as negative. With that, precision and recall are computed based on the number of true positives (TP), false positives (FP), and false negatives (FN). Precision measures how reliable the detections are, indicating out of all positive detetions, how many are true. Recall measures out of all ground truth values, how many are correctly detected. The equations for both precision and recall are given in Equation 3.8 and Equation 3.9 respectively, for a single class detection model.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.9}$$

Precision and recall are both in trade-off: lowering the detection confidence threshold increases the number of accepted detections, which improves recall but typically also increases false positives and reduces precision. Conversely, imposing a stricter confidence threshold increases precision at the cost of recall.

Therefore, Average Precision (AP) taken at a certain threshold, for example AP_{50} evaluates this precision-recall trade-off when predictions are considered correct only if IoU ≥ 0.50. The IoU threshold is first used to determine whether each predicted bounding box is counted as a true or false positive. The predictions are then sorted by decreasing confidence score, and true positives and false positives are accumulated sequentially to compute precision and recall at each step. With that, AP is then obtained as the numerical integral of precision with respect to recall, equal to the area under the resulting precision-recall curve. As it balances precision and recall without requiring a fixed confidence threshold, AP is typically used as the decisive detection evaluation metric.

In addition to AP, the F1-score is used to evaluate the balance between precision and recall at a specific confidence threshold. The F1-score is defined as the harmonic mean of precision and recall, as given in Equation 3.10. Because it penalizes cases where either precision or recall is low, it is useful for selecting a practical operating point for the detector.

$$\mathrm{F1} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{3.10}$$

While AP summarizes detection performance across all confidence thresholds, the F1-score is evaluated at one specific confidence threshold. This makes it suitable for threshold tuning when a single decision threshold must be chosen. In practice, the detector confidence threshold is varied over a predefined range, and for each threshold, precision and recall are computed using Equation 3.8 and Equation 3.9. The corresponding F1-score is then calculated using Equation 3.10. The threshold that maximizes the F1-score is selected as the optimal operating threshold.

Given that, it should be noted that maximizing F1 assumes that precision and recall are equally important. If false positives and false negatives have different operational costs, another threshold selection criterion may be more appropriate depending on the application.

Table 3.2: Detection Performance Metrics, to quantify performance on detection level.

Name	Unit	Description
Precision	-	Ratio of true positives to total detections.
Recall	-	Ratio of true positives to total ground-truth whales.
F1-score	-	Harmonic mean of precision and recall at the selected confidence threshold.
AP_{50}	-	Average Precision at IoU = 0.50.

3.2.3. Whale Monitoring

As section 2.2 showed, AI-based Tip and Cue is most relevant for applications that combine large search areas, high spatial resolution requirements, and sensitivity to response time. Based on the evaluation of candidate applications using the AI-Based Tip and Cue Application Framework presented in Chapter 9, whale monitoring was selected as the demonstration case. Whale presence is widely used as an indicator of marine ecosystem health and biodiversity, making it societally relevant for conservation and environmental monitoring [10].

At the same time, whales are sparsely distributed over large ocean areas and require very high spatial resolution to be distinguished from visually similar ocean features like waves, foam, glare, vessels, or other floating objects. The task is further complicated by whale behaviour, as whales are only visible near the surface for short periods before submerging again, in the order of 8-9 minutes [13]. These spatial and temporal constraints make whale monitoring a suitable case for examining how imaging conditions, detection performance, and response time interact in practice. The detailed application selection methodology is provided in Appendix A.

Deep Learning for Whale Detection

Previous studies have demonstrated that whales can be detected in very-high-resolution satellite imagery using deep learning approaches, including classification, detection, and segmentation models. Early work first demonstrated that whales could be identified in satellite imagery using manual thresholding

methods, but these approaches were not scalable and proved sensitive to waves and water turbidity [88]. Later studies replaced manual procedures with CNN-based classification and detection models, including ResNet, DenseNet, Inception-based networks, Faster R-CNN, and YOLO variants, showing that automated whale detection is feasible in very-high-resolution imagery [11], [43], [89], [90], [91].

Subsequent work also explored segmentation-based approaches including Mask R-CNN and U-Net to obtain more precise whale outlines [92], [93]. Across these studies, the main recurring limitation is that model performance degrades strongly under challenging environmental conditions, including waves, glare, sea state, turbidity, and low contrast [89], [92], [94]. More recent work therefore investigates strategies such as hard-mining, ensembles, and hybrid pipelines to improve robustness, but computational cost and generalization to diverse real-world conditions remain open challenges [95], [96], [97].

An overview of available satellite whale datasets used in this work is provided in Appendix B. Most studies rely on commercial WorldView or GeoEye imagery and provide only image identifiers for purchase, often with restricted access to annotations. These limitations hinder reproducibility and progress in the field [89], [98], [99].

The *Whales from Space* dataset [100] is the first publicly accessible satellite-based whale dataset, providing 633 labelled VHR patches extracted from WorldView-2, WorldView-3, GeoEye-1, and QuickBird-2 imagery. This dataset has become widely used in whale detection research [43], [100], [101], [102].

Although aerial imagery offers higher spatial resolution, down to 10 cm [11], [95], downsampling does not reproduce true orbital radiometry, spectral response, or atmospheric effects.

Taken together, these studies demonstrate that deep learning can detect whales in very-high-resolution satellite imagery, but progress remains constrained by limited data availability given the commercial licensing restrictions and the limited amount of suitable satellite imagery. As a result, many models are trained on relatively small datasets and under favourable environmental conditions, reducing their reliability in operational monitoring scenarios [89], [99], [103].

This limitation directly affects the ability to monitor whale populations at large spatial scales. Without robust detection models and sufficiently diverse datasets, satellite-based whale monitoring cannot yet provide consistent and reliable information on population abundance [103], [104], which is required for assessing ecosystem health and informing marine protection policies [10].

4

Research Definition

This chapter formulates the main research question and defines the sub-questions that guide the analysis in this work.

4.1. Main Research Question

The objective of this thesis is to evaluate the feasibility and performance of AI-Based Tip and Cue formations by jointly considering orbital configuration, off-nadir imaging effects, and onboard detection performance.

The main research question is formulated as:

> **How do orbital configuration and off-nadir imaging geometry influence the detection performance and mission feasibility of AI-Based Tip and Cue systems?**

4.2. Sub Research Questions

To address this question, the following sub-questions are defined.

- **SQ1. Simulation Framework:** How can an integrated benchmark and simulation framework be developed to evaluate AI-Based Tip and Cue systems at both mission and imaging level?
- **SQ2. Imaging Effects:** How do off-nadir viewing angles and associated radiometric distortions affect Cue onboard detection performance for whale detection?
- **SQ3. Mission Geometry:** How do orbital spacing and time delay between Tip and Cue satellites influence latency, viewing geometry, and detection feasibility?
- **SQ4. Comparative Evaluation:** How does the performance of AI-Based Tip and Cue compare to alternative satellite configurations in terms of detection rate, latency, and satellite utilization?
- **SQ5. Application Perspective:** How can a systematic application framework be established to structure and evaluate potential use cases of AI-Based Tip and Cue architectures?

5

Methodology

This chapter presents the methodology used to evaluate AI-Based Tip and Cue systems in relation to the research questions defined in Chapter 4, following the benchmark from subsection 5.1.1.

First, section 5.1 introduces the overall methodological framework and defines how orbital modelling, image generation, and onboard detection are combined into a single evaluation pipeline. Based on this structure, section 5.2 describes the orbital simulation, which provides the satellite-target geometry required for subsequent image generation. These geometric outputs are then used in section 5.3, where physically-based off-nadir images are generated. The resulting images form the input to the onboard detection pipeline described in section 5.4, where detection performance is evaluated. Finally, section 5.5 combines these components into a set of experiments to assess imaging effects, mission geometry, and overall system performance.

5.1. Methodological Framework

To answer the research questions, an integrated simulation and evaluation framework is developed that combines orbital modelling, physically based off-nadir rendering, and onboard detection assessment.

First, a benchmark scenario is defined to enable quantitative comparison between AI-Based Tip and Cue formations and alternative satellite configurations. Performance metrics include detection capacity, observation efficiency, latency, viewing time, and detection performance given off-nadir viewing. This benchmark provides the basis for addressing SQ1 and SQ4.

Second, mission simulation software is implemented to model orbital propagation, relative satellite positioning, and configurable time delay between Tip and Cue satellites. This allows quantification of revisit time, response latency, and achievable off-nadir angles. In addition, the geometric inputs required for off-nadir image generation should be provided by the orbital simulation framework, directly linking mission geometry to image formation and addressing SQ3.

Third, an off-nadir imaging pipeline is developed to simulate geometric distortions, increased ground sampling distance, and radiometric effects like sun glint and angle-dependent reflectance. The resulting images are used to evaluate the impact of viewing geometry on onboard detection performance accross different satellite configurations, in order to answer SQ4.

Finally, an application framework is constructed to systematically structure and assess potential use cases for AI-Based Tip and Cue architectures, which provides context for interpreting the technical findings and supports informed mission design considerations to address SQ5.

The distinct work packages, Work Breakdown Structure, and Work Flow Diagram are provided in Appendix I.

5.1.1. Benchmark Definition

To compare AI-Based Tip and Cue configurations against alternative satellite architectures, a benchmark is defined that fixes the orbital, sensor, and environmental assumptions such that differences in performance

can be attributed primarily to architectural design choices.

Performance Metrics
The benchmark is structured into three groups of metrics. First, Mission Metrics quantify the overall useful mission output in terms of successfully confirmed whales. Second, Response Metrics describe how quickly and with what operational margin these successful confirmations are achieved. Third, Detection Metrics report the standalone performance of the onboard detector on the generated Cue imagery. In this way, the benchmark distinguishes between system-level mission effectiveness, operational responsiveness, and detector quality.

Mission Metrics The main benchmark metric is the successful detection capacity C_{succ}, defined in Equation 5.1. It measures the number of successful whale detections obtained per satellite per simulated day and therefore serves as the overall benchmark score.

$$C_{\text{succ}} = \frac{N_{\text{succ}}}{N_{\text{sat}}\, T_{\text{sim}}} \tag{5.1}$$

Here, N_{succ} is the number of successful whale detections, N_{sat} is the total number of satellites in the configuration, and T_{sim} is the simulation duration in days. A higher C_{succ} indicates that a configuration produces more operationally useful whale confirmations per satellite per day.

A whale detection event is considered successful if it completes the full end-to-end confirmation chain, satisfies the application-specific latency requirement $\tau_i \leq \tau_{\max}$, and contains at least one valid retained detection for that whale in the corresponding Cue image. Since the objective is whale detection, the benchmark operates per ground-truth whale. Consequently, a single Cue image may contribute multiple successful detections if multiple whales are present and validly localized.

Let b_i represent the ground-truth bounding box of whale i, and let $\{\hat{b}_{ij}, s_{ij}\}$ be the set of onboard whale predictions in the corresponding Cue image, where $\hat{b}_{ij}$ is the predicted bounding box and s_{ij} its confidence score.

Only predictions with confidence $s_{ij} \geq s_{\min}$ are considered, where $s_{\min}$ is the deployed confidence threshold selected once during model tuning on the validation folds and then kept fixed for all benchmark experiments.

For comparisons between different optimized detector models, the deployed threshold $s_{\min}$ is selected separately for each model using the same validation protocol. In particular, for each model, the confidence threshold is chosen on the validation folds as the threshold that maximizes the F1-score, after which it is fixed in all benchmark experiments. This ensures that each model is evaluated at its own validation-selected operating point, while maintaining a consistent threshold selection procedure across models.

For whale i, the set of valid detections j is defined in Equation 5.2.

$$D_i = \left\{ j \;\middle|\; s_{ij} \geq s_{\min} \text{ and } \mathrm{IoU}(\hat{b}_{ij}, b_i) \geq \eta \right\} \tag{5.2}$$

η represents the adopted IoU threshold. Thus, D_i contains all retained predictions that localize whale i with sufficient overlap.

The event-level success indicator is then defined in Equation 5.3.

$$m_i = \begin{cases} 1, & \tau_i \leq \tau_{\max} \text{ and } |D_i| \geq 1 \\ 0, & \text{otherwise} \end{cases} \tag{5.3}$$

According to this logic, whale i is counted as successfully detected only if it is confirmed within the required latency and at least one retained prediction j in the Cue image overlaps the corresponding ground-truth whale with IoU above the adopted threshold.

The total number of successful detections is then obtained by summing all successful whale events, as defined in Equation 5.4.

$$N_{\text{succ}} = \sum_{i=1}^{N_{\text{gt}}} m_i \tag{5.4}$$

Here, N_{gt} is the total number of ground-truth whales present in all evaluated Cue images, and N_{succ} is the number of those whales that are successfully confirmed. With this definition, false negatives are included directly in the benchmark, since missed whales, whales with only low-confidence predictions, whales with insufficient overlap, and whales detected too late all yield $m_i = 0$ and do not contribute to N_{succ}. False positives do not increase N_{succ}, since only retained predictions that match a ground-truth whale with sufficient overlap can contribute to Equation 5.4.

Response Metrics The Response Metrics describe the operational quality of the successful detections. These metrics are the mean latency L and the mean viewing time V, defined in Equation 5.5 and Equation 5.6, respectively.

$$L = \frac{1}{N_{\text{succ}}} \sum_{i=1}^{N_{\text{gt}}} m_i \tau_i \tag{5.5}$$

$$V = \frac{1}{N_{\text{succ}}} \sum_{i=1}^{N_{\text{gt}}} m_i v_i \tag{5.6}$$

In Equation 5.5, τ_i represents the latency of whale i, that is, the time between the initial observation opportunity and final successful confirmation. Lower values of L indicate faster response. In Equation 5.6, v_i denotes the remaining viewing time of whale i after first valid Cue acquisition before it exits the allowable off-nadir range. Higher values of V indicate a wider confirmation window and therefore a larger operational margin once the Cue satellite has acquired the target.

Detection Metrics In addition to the mission benchmark, detector-level metrics are reported separately to describe the standalone performance of the onboard detector. These metrics are not included in the mission design comparison, as these do not vary amongst mission architectures, but provide complementary information on detection quality when comparing different models.

The detection metrics consist of mean Average Precision at an IoU threshold of 0.50, denoted as AP_{50}, and the deployed-threshold precision P and recall R. AP_{50} is computed following the COCO evaluation procedure and summarizes detector performance across confidence thresholds under a fixed localization criterion, as described in Figure 3.2.2. In contrast, precision and recall are reported at the fixed deployed threshold $s_{\min}$ used in the benchmark and are defined in Equation 5.7 and Equation 5.8, respectively.

$$P = \frac{TP}{TP + FP} \tag{5.7}$$

$$R = \frac{TP}{TP + FN} \tag{5.8}$$

In Equation 5.7 and Equation 5.8, TP, FP, and FN are the number of true positives, false positives, and false negatives, respectively, under the deployed confidence threshold $s_{\min}$ and IoU threshold 0.5. Higher precision indicates fewer false alarms among reported detections, while higher recall indicates that a larger fraction of whales present in the evaluated Cue images is successfully localized.

Evaluation Procedure
For each satellite configuration, the benchmark follows a simulation pipeline.

1. Propagate satellite orbits using the defined Keplerian elements.
2. Propagate whale targets with given heading and speed dynamics.
3. Evaluate Tip detections using predefined true positive and true negative rates.
4. Apply confirmation delay and task the Cue satellite.
5. Compute the required Euler angles for the Cue satellite and point the sensor toward the target.
6. Generate the corresponding physics-based off-nadir image upon successful observation.
7. Evaluate onboard detection performance.
8. Aggregate mission-level and system-level metrics.

The pseudocode representing this pipeline is provided in Appendix C.

Baseline Scenario
All configurations are evaluated under identical simulation assumptions, summarised in Table 5.1. The benchmark is based on the published orbital and sensor characteristics of WorldView-3 for the Cue satellite. With a nadir ground sampling distance of 0.31 m and a 13.1 km swath [105], WorldView-3 provides sufficient spatial resolution to reliably detect and distinguish individual whales in VHR satellite imagery, as demonstrated in for example the Whales in Space Dataset [100], and is therefore used as the confirmation reference. Note that those can be specified for the field of view parameter only, the input Whales from Space dataset contains images with 0.3 - 0.54 ground sampling distance, which determine the actual dataset generation characteristics as described in subsection 5.4.1.

For the Tip satellite, a Sentinel-2-like sensor is assumed, with a 290 km swath and 10 m ground sampling distance [106]. This resolution is not sufficient for whale confirmation, but it is on the same spatial order as the length of large whales (10-20 m), meaning that surfacing whales may appear as localized pixel-scale anomalies.

Ideally, a slightly finer ground sampling distance, of around 5 m, would allow a whale to span multiple pixels, reducing the risk of confusing true targets with isolated noise or single-pixel artifacts. However, as Tip imagery is not explicitly simulated in this work and the objective is to build on existing sensor characteristics, the Sentinel-2 specifications are used as a representative and operationally realistic reference for the initial benchmark configuration. Mission parameters can later be optimized for deployment-specific studies. With that, the Tip-to-Cue swath ratio of approximately 20 reflects the architectural principle of Tip and Cue: large-area coverage at moderate resolution for anomaly detection, followed by high-resolution targeted imaging for confirmation.

The orbital elements and sensor parameters used in the simulation are listed in Table 5.1. Both satellites share identical orbital elements, with the Cue satellite used as reference building from a Two-Line Element Set (TLE) file at the present date, while the Tip satellite is phase-shifted to realise the defined Tip-Cue delay.

For the baseline scenario, a fixed onboard processing latency of 90 seconds is assumed for the Tip satellite, which has been shown to be achievable in the literature while also accounting for expected future improvements in onboard computational resources [107]. The Cue satellite therefore follows the Tip satellite with a nominal delay of 5 minutes.

The maximum off-nadir angle requirement was derived from the literature review in Figure 3.2.1. At approximately 40° off-nadir, the ground sampling distance increases to about 0.50 m, which was identified as the estimated limit for reliable whale detection.

Table 5.1: Reference conditions for benchmark evaluation.

Category	Parameter	Value
Simulation	Simulation Start date	20/09/2025, 02:33:09 UTC
Scenario	Number of whale targets	500
Simulation	Duration	24 h
Orbit (Cue)	Perigee altitude	615.7 km
Orbit (Cue)	Apogee altitude	624.6 km
Orbit (Cue)	Inclination	97.8703°
Orbit (Cue)	RAAN	336.4191°
Orbit (Cue)	Argument of periapsis	110.0511°
Orbit (Cue)	Mean anomaly	250.1394°
Sensor (Cue)	Swath width	13.1 km
Sensor (Cue)	Nadir GSD	0.31 - 0.54 m
Sensor (Tip)	Swath width	290 km
Sensor (Tip)	Nadir GSD	10 m
Tip-Cue	Nominal time delay	5 min
Tip-Cue	Off-nadir limit	40°
Environment	Wind speed (DEM)	7.5 m/s

All configurations use identical whale seeds, orbital parameters, atmospheric assumptions, and rendering settings to ensure comparability between experiments.

A simulation is performed with 500 whale targets over a 24-hour period. This distribution is not intended to represent real whale population densities, but the targets are propagated to provide a controlled scenario that balances observational sampling with computational feasibility.

Since no dataset was available to train an onboard detection model for the Tip satellite, as discussed in Table 5.4.1, the detection performance of the Tip satellite cannot be modeled explicitly. Therefore, Tip detections are simulated using a stochastic detection model with a True Positive Rate and True Negative Rate of 0.85, compatible with prior whale detection model capabilities [11], [89]. Detection outcomes are sampled according to these rates, while the resulting observations are subsequently evaluated using the Cue satellite detection model.

5.1.2. Software Architecture

The goal of the simulation framework is to evaluate AI-Based Tip and Cue mission configurations under a predefined satellite architecture and a maximum off-nadir angle requirement. For a given scenario, the framework will derive the satellite-target geometric parameters required to generate off-nadir imagery and assess detection performance. The framework also calculates the latency between Tip detection and Cue observation, as well as the available viewing time per target.

These parameters are then used to determine the total number of successfully observed targets for each mission configuration. This enables a direct comparison between architectures, for example between a multi-satellite Cue constellation and a paired Tip and Cue system, following the benchmark definition in subsection 5.1.1.

To develop this simulation framework, the software was divided into four modules: Orbital Simulation, Off-Nadir Imaging, Onboard Detection, and Evaluation, as shown in Figure 5.1.

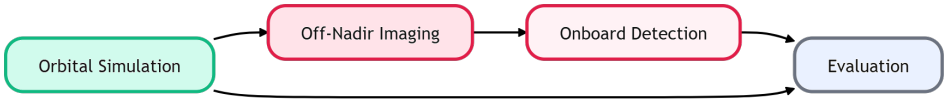


Figure 5.1: High-level software block diagram of the simulation framework, showing the interaction between orbital simulation, off-nadir imaging, onboard detection, and evaluation modules.

These modules provide inputs for each other. The orbital simulation framework propagates the targets and Tip and Cue satellites in orbit. Then it simulates 'detections', after which the geometric configuration

of the detection is passed on to Off-Nadir Imaging. The Off-Nadir Imaging pipeline generates a physically-representative off-nadir image for that specific geometry, which is then passed to the Onboard Detection module to evaluate onboard whale detection performance. Both parameters, like latency and viewing time from the simulation framework, and onboard detection performance, are logged, so that the mission can be evaluated following the benchmark.

Mission Simulation Framework
Following the high-level implementation, the expansion for the Orbital Simulation Framework is given in Figure 5.2. It shows how a complete Tip and Cue mission scenario is modeled, as detailed in section 5.2.

In Initialization, the simulation is initialized: setting up directories, initializing Orekit, and PASEOS simulation. Next, in Constellation & Actor Setup, a constellation of N orbits with M satellites per orbit is initialized. After that, in Mission Tools, the attitude controllers and Earth observation modules are linked to each satellite.

Then in Target Environment, K targets are initialized and propagated over time. In Simulation Core loop, satellites are propagated in orbit and their projected fields of view are evaluated at each timestep. If a target falls within the Tip satellite footprint, checked in Tip Detection Pipeline, it is forwarded to the Cue satellite after a specified processing delay.

In Cue Tasking & Pointing, the Cue satellite maintains a task list of candidate targets and selects the task that remains feasible when accounting for its pointing and stabilization time as well as the maximum off-nadir constraint. After selecting a task, the Cue satellite slews its boresight towards the target by modeling roll, pitch, and yaw dynamics, including rotational accelerations. Once stabilized, in Cue Observation, the target is observed when it enters the Cue footprint. The task is then removed from the queue and the Cue satellite becomes available for subsequent assignments.

During each observation, the satellite and target latitude, longitude, altitude, and time are logged. These quantities are used as inputs to the Off-Nadir Imaging Pipeline described in section 5.3, where off-nadir images are generated. The image serves as input for Onboard Detection, where the detection performance is evaluated as detailed in section 5.4. To enable simulations that only evaluate Mission- and System-Level metrics, a bypass to Cue Onboard Detection is created, using a dummy model to simulate latency. Geometric parameters are still logged, allowing for the evaluation of Onboard Detection at a later stage.

Next, all parameters, like the latency between Tip and Cue observations and the viewing time, defined as the time interval between Cue observation and the moment the target exits the Cue field of view, are recorded. In addition to geometrical configuration, including latitude, longitude, and altitude of both Target and Satellite at detection, time, and Cue onboard detection performance are logged in Logging & Metrics.

After finalizing the mission, those are converted in Mission Evaluation to the final mission overview in Excel, footprint visualizations, and a 3D simulation video in the Outputs folder.

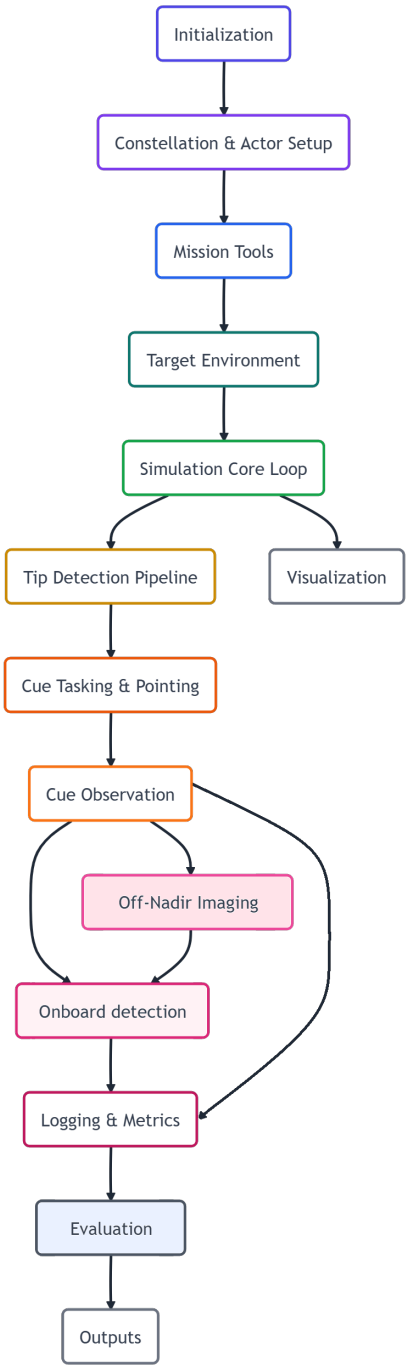


Figure 5.2: Detailed workflow of the Tip and Cue simulation framework, illustrating the sequence from mission simulation and tasking through off-nadir image generation, onboard detection, and performance evaluation.

Appendix C provides a detailed expansion, including Python function interactions. It also includes the expansion of the Off-Nadir Image Rendering pipeline, to generate an off-nadir image for a given orbital configuration, and cross validation pipeline to training the detection model.

The simulation framework code is available open source at: `https://github.com/ESA-PhiLab/AI-Based-Tip-and-Cue`.

5.1.3. File Structure

After the simulation is completed, a results folder is automatically generated with the outputs as listed in Table 5.2.

Table 5.2: File structure of the results folder after simulation completion.

File Name	Description
`footprints_cue_<run_name>.html`	An interactive representation of the Cue satellite footprints.
`footprints_tip_<run_name>.html`	An interactive representation of the Tip satellite footprints.
`logs_<run_name>.txt`	A log file capturing execution and status information during the simulation.
`plot_gsd_<run_name>.png`	A plot showing Ground Sampling Distance (GSD) distribution.
`plot_latency_confirmation_<run_name>.png`	A plot showing the confirmation latency distribution.
`plot_latency_observation_<run_name>.png`	A plot showing the observation latency distribution.
`plot_offnadir_<run_name>.png`	A plot showing off-nadir angle distribution.
`plot_viewing_time_<run_name>.png`	A plot showing the viewing time distribution.
`results_<run_name>.xlsx`	An Excel file containing the summarized results of the orbital simulation.
`settings.py`	Python file containing the configuration settings used for the simulation.
`satellite_config.py`	Python file containing the satellites orbit and sensor specifications.
`mov_<run_name>.mov`	A movie file capturing the simulation's visual output.
`satellite_images`	Off-nadir images at the specific target detection instances.
`supporting_dataset`	Additional dataset including different processing stages (reflectance and radiance), nadir geometry, and with or without sunglint simulation for the same imaging geometry.
`onboard_detection`	Folder containing Excel overviews and bounding box images with the detection results.
`benchmark_<run_name>.xlsx`	An Excel file containing all the benchmark performance metric scores for the simulation run.

The parameters from the benchmark, including detection coverage, latency, off-nadir angle, and viewing time, are stored in the `results_<run_name>` Excel file, which can be extracted in the evaluation to compare different mission configurations.

5.1.4. Computational Environment

To execute the simulation, this section covers the software tools and methods used for mission simulation. Hardware tools are described in subsection 5.1.4, and software tools are described in Table 5.1.4.

Hardware

Computational experiments were conducted across three computing environments during the project. Until December 2025, a remote ESA ϕ-lab machine was used for environment testing and early pipeline development. After access expiration, all model training and large-scale evaluation runs were executed on the TU Delft MSc Robotics *iv-mind* server.

Dataset generation and mission simulation experiments were performed locally. These runs were executed on an HP ZBook Studio G5 (Intel i7-7700HQ, 4 cores / 8 threads, NVIDIA Quadro M1200 with 4 GB VRAM) until November 2025. After that, the local device was upgraded to HP OMEN MAX 16-ah0670nd (Intel Core Ultra 7 255HX, 20 cores, NVIDIA GeForce RTX 5080 with 16 GB VRAM).

To ensure reproducibility and transparency, the main hardware characteristics of the ESA ϕ-lab machine and the TU Delft *iv-mind* server are summarised in Table 5.3, and for the local machine in Table 5.4.

Table 5.3: Hardware characteristics of the remote computing infrastructure used for model fine-tuning and evaluation during the project.

Component	ESA ϕ-lab Machine	TU Delft iv-mind Server
CPU	AMD EPYC (4 cores)	2× Intel Xeon E5-2690 v4 (56 threads total)
RAM	14 GB	251 GB
GPU	NVIDIA L40S-6Q (6 GB VRAM)	8× NVIDIA Tesla V100-SXM2 (32 GB each)
CUDA Version	12.2	13.0 (Driver), 12.4 (PyTorch runtime)
PyTorch	2.2.2 + cu121	2.5.1 + cu124
Storage	256 GB SSD	28 TB system + 6.6 TB RAID scratch
Operating System	Linux	Ubuntu 22.04.5 LTS

Table 5.4: Local computing devices used for dataset generation and mission simulation.

Component	HP ZBook Studio G5 (until Nov 2025)	HP OMEN MAX (after Nov 2025)
CPU	Intel i7-7700HQ (4 cores / 8 threads)	Intel Core Ultra 7 255HX (20 cores)
RAM	16 GB	16 GB
GPU	NVIDIA Quadro M1200 (4 GB VRAM)	NVIDIA GeForce RTX 5080 (16 GB VRAM)
CUDA Version	12.8 (driver-reported)	13.1 (driver-reported)
Operating System	Windows 10 + WSL2 Ubuntu	Windows 11 + WSL2 Ubuntu 24.04 LTS
Storage	256 GB SSD	2 TB SSD (system), 3.7 TB external

Software

The software environment was structured following the functional components of the AI-Based Tip and Cue framework: Orbital Simulation, Off-Nadir Imaging, Onboard Detection, and Development & Documentation. An overview is provided in Table 5.5.

Table 5.5: Software tools used in the AI-Based Tip and Cue framework, grouped by functional component.

Category	Tool	Purpose	Functions
Orbital Simulation	Python	Core simulation framework	Integration of orbital, rendering, and detection modules
	OREKIT	Orbital mechanics library	Walker constellation generation, orbital propagation
	PASEOS	Spacecraft simulation framework	Satellite object definition and state propagation
Off-Nadir Imaging	Mitsuba	Physically-based renderer	BRDF modelling, ray tracing, image rendering
	SMARTS	Radiative transfer model	Solar spectral irradiance computation
	NumPy / SciPy	Geometry processing	DEM generation, coordinate transforms
Onboard Detection	PyTorch	Neural network framework	Model training and inference
	DEIMv2	Object detection model	Whale detection fine-tuning
	Bash	Cross-validation automation	Fold training and GPU distribution
	COCO API	Dataset analysis	mAP, precision, recall computation
	TensorBoard	Visualization tool	Training performance monitoring
Development & Documentation	Git	Version control	Code management and reproducibility
	PyCharm	Python IDE	Development and debugging
	Visual Studio Code	LaTeX editing	Thesis writing and documentation

All modules were implemented in Python and designed to operate modularly. Orbital simulation (OREKIT, PASEOS) generates geometric inputs for the off-nadir image rendering pipeline (Mitsuba, SMARTS), whose outputs are directly passed to the onboard detection module (DEIMv2), ensuring consistency between mission geometry, radiometric modelling, and onboard detection evaluation.

5.2. Orbital Simulation

This section describes the orbital simulation framework developed for AI-based Tip and Cue Earth observation missions. It first presents the simulation setup, including target propagation, constellation setup, and orbital propagation. Next, the observation geometry is introduced through the reference frames, attitude rotation, and field-of-view projection. The Tip and Cue tasking logic is then described, including the illumination constraints and task selection procedure. After that, the pointing and stabilization model is presented through the slew maneuver, stabilization phase, and its execution in the simulation. Finally, the data logging and state handling are explained.

5.2.1. Simulation Setup

Target Propagation

To set up a simulation scenario, a procedure to generate and propagate artificial targets was defined in order to evaluate Cue tasking under estimated target motion. Given whale targets, its initialization is restricted to ocean locations using the GSHHG shoreline data. Specifically, the GSHHG L1 layer is used to represent the global land-ocean boundary, excluding Antarctica, and L5 is added to include Antarctica [108], [109]. The two layers are merged and rasterized to a binary land mask based on latitude-longitude coordinates. Random targets are then sampled uniformly on the sphere and accepted only if they fall on water pixels of this mask. To avoid unrealistic polar clustering and to automatically exclude Arctic and Antarctic land regions, sampling is restricted to a maximum geodetic latitude of 70 degrees.

To account for the latency between Tip and Cue, the framework must capture whether a target remains within the Cue observation window after it was first detected and tasked. Each target is therefore also

assigned an initial speed and heading. The heading is sampled uniformly from $[0°, 360°)$, and the speed is sampled from $v \in [0.2, 6.0]$ m/s), with a mean centered at 1.5 m/s [110] [111] [112]. Target motion is updated at each simulation timestep Δt by first updating the speed and heading, and then propagating the position along its spherical distance.

Let (φ_k, λ_k) be the latitude and longitude at timestep k, v_k the speed, and ψ_k the heading, bearing clockwise from north. The traveled distance over one timestep is then equal to $d_k = v_k \Delta t$.

Assuming a spherical Earth with radius $R_\oplus$, the angular distance can for small angles be approximated by $\delta_k = d_k / R_\oplus$. The position is propagated using a forward great-circle geodesic update can be computed with Equation 5.9 and Equation 5.10 for latitude and longitude respectively [113].

$$\varphi_{k+1} = \arcsin\Big(\sin\varphi_k \cos\delta_k + \cos\varphi_k \sin\delta_k \cos\psi_k \Big), \tag{5.9}$$

$$\lambda_{k+1} = \lambda_k + \operatorname{atan2}\Big(\sin\psi_k \sin\delta_k \cos\varphi_k,\ \cos\delta_k - \sin\varphi_k \sin\varphi_{k+1} \Big). \tag{5.10}$$

Target speeds and headings are perturbed to avoid perfectly straight trajectories. The speed follows a mean-reverting update with additive Gaussian noise, while the heading is perturbed by a Gaussian turn increment with variance proportional to Δt. This produces smooth trajectories with stochastic variation in both speed and direction, which is sufficient for evaluating whether targets drift out of the Cue viewing window during Tip-Cue delays.

If a propagated position falls on land, according to the land mask, a land-avoidance rule is applied. The heading is perturbed by a random deflection, the attempted step length is reduced, and the propagation is retried. If no valid water position is found after a fixed number of retries, the target remains at its previous location for that timestep. An additional latitude constraint is enforced: if $|\varphi_{k+1}| > \varphi_{\max}$, the heading is reversed and the step is recomputed to keep targets within the allowed latitude band. This ensures targets remain on water while still producing continuous motion suitable for latency and task-feasibility analysis.

Figure 5.3 shows an example distribution of $N = 500$ initialized targets over the ocean. Figure 5.4 illustrates example target trajectories over time, including cases where motion is adjusted near coastlines by the land-avoidance rule. It is observed that the target crosses the land within the 8-hour propagation patch. This occurs due to difference between the Plotly map and the GSHHG landmasks. However, since Plotly is only used for visualization, and it was confirmed that the whale does not move further inland, this discrepancy does not affect the actual mission simulations.

With different random seeds, targets are initialized at different ocean locations and assigned different initial speeds and headings. While this approach does not reproduce true global whale density patterns, it produces a robus sampling that supports consistent comparison of performance metrics across simulation runs and mission configurations.

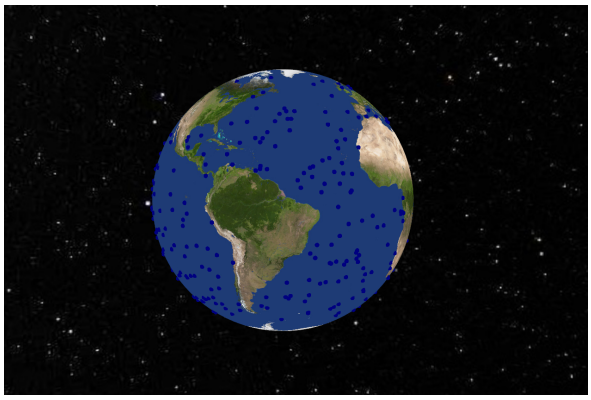


Figure 5.3: Example initialization of $N = 500$ targets sampled on ocean pixels of the GSHHG-based land mask.

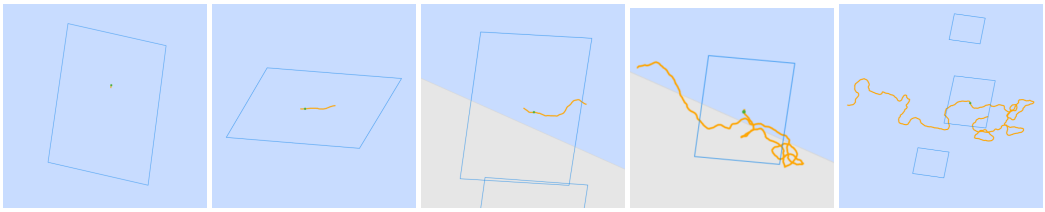


Figure 5.4: Example propagated target trajectories over time. From left to right, propagation time: 5 min, 30 min, 1 hour, 8 hours, 24 hours. The coastline constraint prevents targets from crossing the land defined by the GSHHG masks, where the Plotly map, proving small inaccuracies in the 8-hour patch, is only used for visualization.

Constellation Setup

After having defined an Earth model with targets, the option to build a constellation of satellites was implemented. First, the Keplerian orbital parameters of a reference Cue satellite are specified (semi-major axis a, eccentricity e, inclination i, right ascension of the ascending node Ω, argument of perigee ω, and mean anomaly M). Around this reference orbit, a Walker-Delta constellation is generated based on a predefined number of orbital planes and number of satellites per plane.

The Walker-Delta constellation was chosen over the Walker-Star because it has been shown that, for Sun-synchronous orbits at 600 km and 800 km altitude with inclinations of approximately $97.72°$ and $98.53°$, respectively, the Delta configuration provides improved coverage and revisit performance at higher latitudes for remote sensing missions [114].

In a Walker-Star constellation, orbital planes are distributed over $180°$ in right ascension, whereas in a Walker-Delta constellation they are distributed over $360°$, as illustrated in Figure 5.5. As a result, Walker-Star constellations intersect at the poles, while Walker-Delta constellations show a uniform longitudinal distribution of ground tracks and inter-plane crossings.

For the Tip and Cue architecture, inter-satellite communication between neighbouring planes may also be desired, for example to enable task handover between different Cue satellites within the same geographical region. In Walker-Star configurations, inter-satellite links are typically limited to satellites within the same orbital plane [115]. Therefore, the Walker-Delta configuration was chosen, given higher flexibility for cross-plane communication and improved coverage, which is beneficial for the distributed Tip and Cue concept.

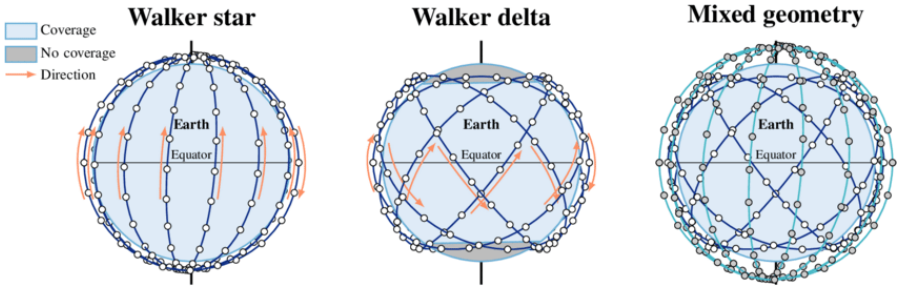


Figure 5.5: Comparison of Walker star, Walker delta, and mixed constellation geometries, showing their different orbital-plane distributions and resulting global coverage patterns. [116]

To generate the constellation, let P represent the number of planes and S the number of satellites per plane, so that the total number of satellites equals $T = PS$. The orbital planes are evenly distributed in right ascension so that the right ascension of the ascending node (RAAN) of plane p is given by Equation 5.11 [117].

$$\Omega_p = \frac{360°}{P}(p-1). \tag{5.11}$$

Within each plane, satellites are evenly spaced in mean anomaly as given by Equation 5.12.

$$\Delta M_{\text{in-plane}} = \frac{360°}{S}. \tag{5.12}$$

In addition, a Walker phasing parameter $F \in \{0, \ldots, P-1\}$ was introduced to generate an inter-plane along-track offset, to ensure that satellites from different planes are not aligned at identical along-track positions. The mean anomaly of satellite (p, s) is therefore defined in Equation 5.13 [117].

$$M_{p,s} = \frac{360^\circ}{T} F\,(p-1) + \frac{360^\circ}{S}\,(s-1). \tag{5.13}$$

The first term in Equation 5.13 shifts neighbouring planes relative to each other by $\frac{360^\circ}{T}F$ in mean anomaly, while the second term distributes satellites uniformly within each plane.

Next, the position of the Tip satellite is defined relative to the Cue satellite. By specifying a time delay Δt_{TC} between Tip and Cue, the Tip satellite is placed ahead of the Cue along the same orbit. Using the orbital mean motion $n = 2\pi/T_{\mathrm{orb}}$, this corresponds the a mean anomaly offset in Equation 5.14.

$$\Delta M = n\,\Delta t_{\mathrm{TC}}, \tag{5.14}$$

so that the Tip mean anomaly is initialized as $M_{\mathrm{Tip}} = M_{\mathrm{Cue}} + \Delta M$.

Based on the specified number of planes and satellites, a separate Tip constellation can be generated. If the Tip configuration matches the Cue configuration (same P and S), each Tip satellite is placed in front of its corresponding Cue satellite with the same time delay, resulting in a defined number of Tip-Cue pairs.

The Tip location is always derived with respect to the Cue orbit, which also allows simulation of Cue-only constellations without a Tip satellite. In that case, no tasking or handover is performed and targets are only detected if they fall directly inside the Cue footprint at the observation time.

Examples of generated Walker-Delta constellations for both Tip and Cue pair and VHR constellation only are displayed in Figure 5.6 and Figure 5.7 respectively.

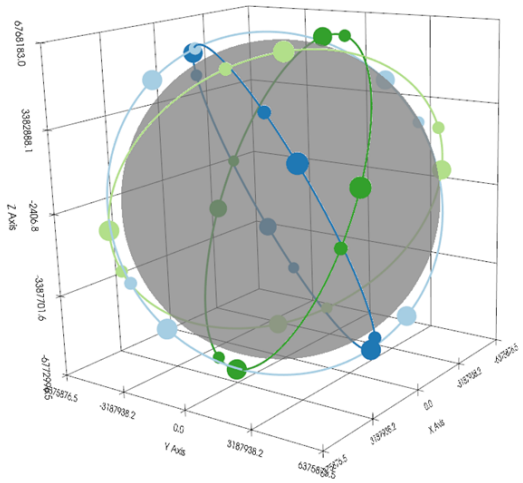


Figure 5.6: A Tip and Cue Walker-Delta constellation of four planes, four T/C satellite-pairs per plane, at 60-degree inclination. .

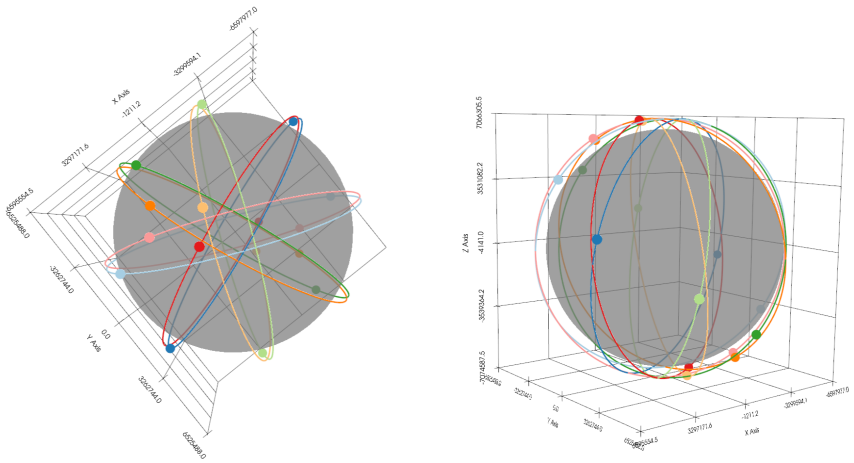


Figure 5.7: A VHR Cue Walker-Delta constellation of eight planes, two satellites per plane, at 97- degree inclination.

Orbital Propagation

Given the initial position and velocity of each satellite from the constellation definition (Figure 5.2.1), a satellite object is instantiated for every node in the simulation. The framework is built on PASEOS [118], which represents each spacecraft as an actor to keep track of their state during the simulation.

Then the orbital motion is propagated using the Orekit numerical propagation framework [119]. For each satellite, an Orekit propagator is initialized from Keplerian elements at a common epoch, and then advanced to obtain the spacecraft state, consisting of position and velocity, at the successive simulation time.

It should be noted that PASEOS also supports additional operational and onboard constraint models, for example ground station communication windows, power, and thermal states, but these were not enabled as they are out of scope for this project. PASEOS was primarily used to enable software expansion within the existing ESA framework, and two mission-specific modules were newly developed:

- **Earth Observation (EO):** responsible for managing all geometry related to the sensor footprint. This includes generating and updating the projected field-of-view on the Earth's surface, track-keeping of the footprint state, and evaluating whether targets fall within the observable region (subsection 5.2.2).
- **Attitude:** responsible for managing the spacecraft orientation. This includes computing the sensor pointing direction, planning and executing offnadir slewing manoeuvres towards tasked targets, and ensuring stabilization before an observation is performed (subsection 5.2.4).

While PASEOS also provides built-in 3D visualization capabilities, these were neither used. Instead, a visualization module using PyVista and Plotly was newly developed, as the existing plotting tools were not compatible with the integration of target states and projected footprint displays.

5.2.2. Observation Geometry

After defining the satellite objects and their state vectors, a sensor viewing window is required to determine whether a target can be observed. This module was developed in collaboration with Tobia Armando La Marca (Ph.D. in Cosmology, Space Science and Space Technology, ESA ϕ-lab), who provided the initial scripts for field-of-view projection under specified Euler angles and the 2D footprint visualization. These scripts were then corrected, extended, and integrated into the AI-Based Tip and Cue simulation framework.

Reference Frames

From the simulation, the satellite state vectors $\mathbf{r}$ and $\mathbf{v}$ are known. These are expressed in the Earth-centred inertial (ECI) frame, also referred to as the inertial reference frame (IRF). In this work, the IRF corresponds to the EME2000 frame used by Orekit [119].

To determine the sensor footprint, the inertial state must first be transformed into a local satellite reference

frame. Based on the satellite attitude, the sensor viewing directions are then rotated and subsequently projected back into the inertial frame to compute their intersection with the Earth surface.

First, describe sensor pointing in this local satellite frame, the local vertical local horizontal (LVLH) frame is defined. This local reference frame, centered at the satellite, is specified so that the $\hat{\mathbf{z}}_{\text{LVLH}}$ axis points towards the Earth center (nadir direction). The $\hat{\mathbf{y}}_{\text{LVLH}}$ axis is perpendicular to the orbital plane and therefore orthogonal to both the position vector $\mathbf{r}$ and velocity vector $\mathbf{v}$, and the $\hat{\mathbf{x}}_{\text{LVLH}}$ axis completes the right-handed coordinate system. The frame is illustrated in Figure 5.8 [120].

Following the definition, the LVLH unit vectors are constructed from the satellite state vectors $\mathbf{r}$ and $\mathbf{v}$ as in Equation 5.15 [121].

$$\begin{aligned} \hat{\mathbf{z}}_{\text{LVLH}} &= -\frac{\mathbf{r}}{\|\mathbf{r}\|}, \\ \hat{\mathbf{y}}_{\text{LVLH}} &= -\frac{\mathbf{r}\times\mathbf{v}}{\|\mathbf{r}\times\mathbf{v}\|}, \\ \hat{\mathbf{x}}_{\text{LVLH}} &= \hat{\mathbf{y}}_{\text{LVLH}}\times\hat{\mathbf{z}}_{\text{LVLH}}. \end{aligned} \tag{5.15}$$

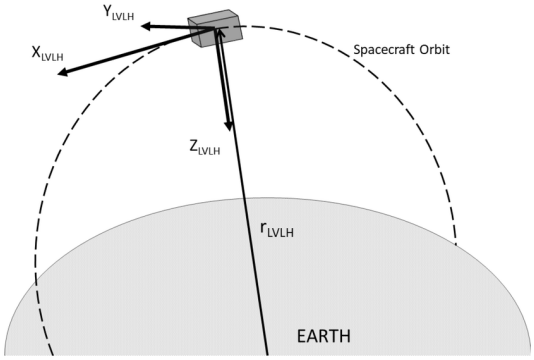


Figure 5.8: Definition of the LVLH reference frame centred at the satellite. [120]

The transformation matrix from IRF to LVLH is then given in Equation 5.16 [122].

$$\mathbf{R}_{\text{IRF}\to\text{LVLH}} = \begin{bmatrix} \hat{\mathbf{x}}_{\text{LVLH}}^{\top} \\ \hat{\mathbf{y}}_{\text{LVLH}}^{\top} \\ \hat{\mathbf{z}}_{\text{LVLH}}^{\top} \end{bmatrix}. \tag{5.16}$$

Attitude Rotation

To go from the nadir-pointing configuration to an off-nadir pointing configuration, it is necessary to convert the nadir-aligned LVLH frame to the body reference frame (BRF), which represents the actual sensor orientation. The attitude with respect to the nadir direction is defined using Euler angles roll (ϕ), pitch (θ), and yaw (ψ).

Using these Euler angles, the LVLH frame can be rotated into the BRF using the rotation matrix given in Equation 5.17.

$$\mathbf{R}_{\text{LVLH}\to\text{BRF}} = \mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\phi), \tag{5.17}$$

where the standard rotation matrices are given in Equation 5.18.

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix},$$
$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \tag{5.18}$$
$$\mathbf{R}_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Field of View Projection

Footprint Ray Definition Given the satellite position and orientation, the footprint is defined as the intersection of a pyramidal field of view, originating from the sensor, with the WGS-84 reference ellipsoid Earth.

As the boresight direction is aligned with the positive z-axis in the body reference frame (BRF), the four corner rays are geometrically computed with Equation 5.19 and normalized in Equation 5.20. Here, θ_x and θ_y represent the full field-of-view angles of the sensor in two orthogonal directions.

$$\tilde{\mathbf{u}} = \begin{bmatrix} \pm \tan\left(\frac{\theta_x}{2}\right) \\ \pm \tan\left(\frac{\theta_y}{2}\right) \\ 1 \end{bmatrix} \tag{5.19}$$

$$\mathbf{u}_i^{\text{BRF}} = \frac{\tilde{\mathbf{u}}_i}{\|\tilde{\mathbf{u}}_i\|}. \tag{5.20}$$

A ray defined in BRF is then transformed back to the inertial reference frame (IRF), which corresponds to the Earth-centred inertial frame (ECI), through the inverse transformations as given in Equation 5.21. This results in the unit vector $\mathbf{u}^{\text{IRF}}$ representing the ray direction expressed in the IRF.

$$\mathbf{u}^{\text{IRF}} = \mathbf{R}_{\text{IRF}\to\text{LVLH}}^{\top}\mathbf{R}_{\text{LVLH}\to\text{BRF}}^{\top}\mathbf{u}^{\text{BRF}}. \tag{5.21}$$

Ray-Ellipsoid Intersection To find the intersections of the new unit vectors, the Earth is modelled as a biaxial ellipsoid with semi-major axis a and semi-minor axis b as in Equation 5.22 [123].

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} + \frac{z^2}{b^2} = 1. \tag{5.22}$$

A ray originating at satellite position $\mathbf{r}_0$ with direction $\mathbf{d}$ is parameterized in Equation 5.23, with its scalar version in Equation 5.24.

$$\mathbf{r}(t) = \mathbf{r}_0 + t\,\mathbf{u}^{\text{IRF}}, \tag{5.23}$$

$$\begin{aligned} x(t) &= x_0 + tu_x, \\ y(t) &= y_0 + tu_y, \\ z(t) &= z_0 + tu_z. \end{aligned} \tag{5.24}$$

Substituting Equation 5.24 into Equation 5.22 results in a quadratic equation of the form $At^2 + Bt + C = 0$, which can be solved numerically to obtain the parameter t.

Once t is determined, and with the field-of-view ray vector $\mathbf{u}^{\text{IRF}} = (u_x, u_y, u_z)$ and the satellite position $\mathbf{r}_0 = (x_0, y_0, z_0)$ known, the intersection point in Cartesian ECI coordinates is computed by substituting these values into Equation 5.24. This intersection point is then transformed from ECI to Earth-fixed coordinates and subsequently converted to geodetic latitude and longitude. Repeating this procedure for all four corner rays yields the footprint polygon. A target at geodetic coordinates (φ, λ) is then considered observable if its latitude and longitude fall within the projected footprint.

To visualize the footprints, the time evolution of the projected field of view along the orbit is shown in Figure 5.9.

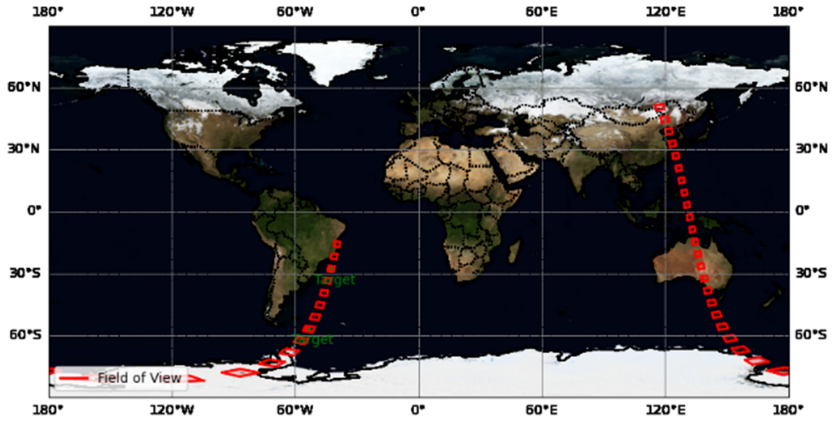


Figure 5.9: Visualization of the orbital propagation of field-of-view windows from the Tip satellite, where the Cue satellite is tasked after the target coincides with the window.

5.2.3. Tip and Cue Tasking Logic

Illumination Constraints

Before a target can be processed by either Tip or Cue, illumination constraints are evaluated for both the satellite and the target. A satellite at position $\mathbf{r}$ is considered to be in eclipse, and therefore excluded, when the conditions in Equation 5.25 are satisfied. Here, $\hat{\mathbf{s}}$ represents the unit vector pointing from the Earth toward the Sun, and $R_\oplus$ is the Earth radius. The Sun-Earth distance is assumed sufficiently large so that solar rays can be treated as parallel.

As illustrated in Figure 5.10, the first condition ensures that the satellite is located on the night side of the Earth. This is the case when the angle between $\mathbf{r}$ and $\hat{\mathbf{s}}$ exceeds $90°$, resulting in a negative dot product [124]. The second condition evaluates the perpendicular distance of the satellite to the Sun-Earth axis. Since $\hat{\mathbf{s}}$ is a unit vector, this distance is given by $\|\mathbf{r} \times \hat{\mathbf{s}}\|$. If this distance is smaller than the Earth radius, the satellite lies within the cylindrical shadow region.

$$\mathbf{r} \cdot \hat{\mathbf{s}} < 0 \quad \text{and} \quad \|\mathbf{r} \times \hat{\mathbf{s}}\| < R_\oplus, \tag{5.25}$$

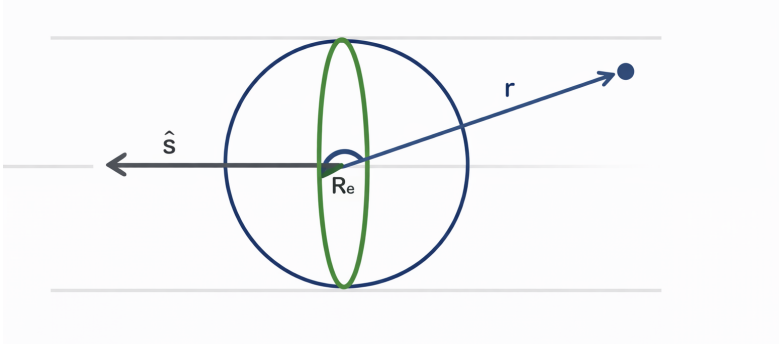


Figure 5.10: Sun-Earth geometry illustrating the illumination constraint and satellite line-of-sight vector. A satellite at position $\mathbf{r}$ is in shadow when the angle with the sun vector $\hat{\mathbf{s}}$ is larger than 90 degrees, and its 3D position exceeds the Earth radius $R_\oplus$.

Targets are filtered using a daylight mask instead. A target located at position $\mathbf{r}_t$ in ECEF is considered illuminated if Equation 5.26 holds, which corresponds to a positive solar elevation angle.

$$\frac{\mathbf{r}_t}{|\mathbf{r}_t|} \cdot \hat{\mathbf{s}} > 0, \tag{5.26}$$

This exclusion of shadowed satellites and nighttime targets avoids spending computation on objects that cannot observe, or be optically observed. The result is shown in Figure 5.11.

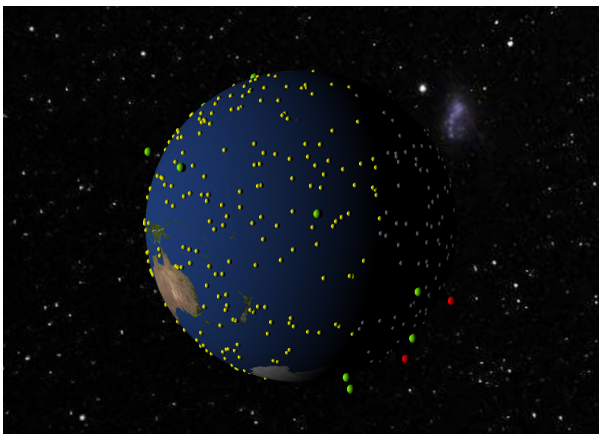


Figure 5.11: Illumination filtering example. Targets and satellites on the daylight side are evaluated for observation (green), while objects located in Earth's shadow are excluded from further processing (grey).

Tasking

To determine whether a target is tasked, for each illuminated Tip satellite-target pair it is evaluated whether the target lies within the projected field-of-view (FoV) footprint. For the Tip satellite, a simplified detection model is adopted. As no suitable high-resolution Tip dataset is available, as discussed in section 5.4, a dummy classifier is introduced with $\mathrm{TPR}_{\mathrm{Tip}} = \mathrm{TNR}_{\mathrm{Tip}} = 0.85$, representative of whale detection performance reported in literature [43], [91], in addition to a fixed onboard processing delay of 90 seconds [107].

If the Tip confirms a positive detection, the detected target is inserted into a global shared Cue task pool together with its identifier, observed coordinates, and assignment time. It becomes available for execution after the Tip-Cue transmission delay Δt_{TC} has elapsed.

Once the transmission delay has passed, any Cue satellite that is not currently executing an active task evaluates the shared task pool. To reduce computational cost, task selection is performed in two stages. First, a coarse geographic prefilter is applied to discard tasks that are unlikely to become observable based on the instantaneous latitude-longitude separation between the Cue subsatellite point and the target location. A task i is retained only if it satisfies Equation 5.27.

$$|\Delta\phi_i| \le \Delta\phi_{\max}, \qquad |\Delta\lambda_i| \le \Delta\lambda_{\max}, \tag{5.27}$$

Here, $\Delta\phi_i$ and $\Delta\lambda_i$ are the latitude and longitude differences between the satellite subsatellite point and the target location, respectively.

Only tasks satisfying Equation 5.27 are subjected to the more expensive propagation-based visibility and slew-feasibility analysis. For each candidate task i, the scheduler determines the earliest feasible future observation time $t_i^\star$. The task with the smallest predicted time-to-observation among all feasible candidates is then claimed by the satellite. Tasks for which no feasible observation opportunity exists are excluded from the selection process. The optimal task index $i^\star$ is therefore obtained using Equation 5.28.

$$i^\star = \arg\min_i \; t_i^\star, \tag{5.28}$$

To determine $t_i^\star$, the scheduler propagates the Cue spacecraft state forward in time and evaluates a sequence of candidate future instants. At each candidate time t_k, the target must first satisfy the line-of-sight visibility condition and the off-nadir constraint. The off-nadir feasibility check is given in Equation 5.29, where $\theta_{\text{off}}(t_k)$ is the angle between the target line-of-sight vector and the spacecraft nadir direction in the LVLH frame, $\theta_{\max}$ is the nominal off-nadir limit, and $\Delta\theta$ is an additional margin used during task evaluation.

$$\theta_{\text{off}}(t_k) \leq \theta_{\max} + \Delta\theta, \tag{5.29}$$

If this condition is satisfied, the normalized boresight direction toward the target is computed as in Equation 5.30.

$$\mathbf{b}_{\text{target}}(t_k) = \frac{\mathbf{r}_t(t_k) - \mathbf{r}_s(t_k)}{\|\mathbf{r}_t(t_k) - \mathbf{r}_s(t_k)\|}, \tag{5.30}$$

Here, $\mathbf{r}_t(t_k)$ is the target position and $\mathbf{r}_s(t_k)$ the spacecraft position at time t_k. This line-of-sight vector is transformed to the LVLH frame, after which the Euler attitude required to align the sensor boresight with the target is determined as described in subsection 5.2.2.

Only geometric feasibility is not sufficient. For every candidate time t_k, the scheduler also evaluates whether the spacecraft can complete the required slewing and stabilization before the target reaches the corresponding observable geometry. This feasibility condition is expressed in Equation 5.31.

$$t_{\text{point}} \leq t_k - t, \tag{5.31}$$

Here, t is the current simulation time and t_{point} is the required slew-plus-stabilization time obtained from the attitude model described in subsection 5.2.4. The earliest candidate time satisfying both Equation 5.29 and Equation 5.31 is taken as $t_i^\star$.

For the selected task $i^\star$, the scheduler stores the corresponding target Euler attitude, the expected time-to-observation, and the associated predicted off-nadir angle. The spacecraft is then commanded directly toward this attitude so that the maneuver is completed when the target becomes observable.

Once a Cue satellite has claimed a task, that task is reserved for that satellite and is removed from consideration by the other Cue satellites. While a task is active, the spacecraft does not switch to another task. If the currently claimed target is found to be no longer observable within the allowed future search horizon, or if the predicted observation time is exceeded without a successful observation, the task is released back into the global shared task pool and the scheduling procedure is repeated. This allows another Cue satellite to claim the same task later if its geometry becomes favorable. When no active task remains, the spacecraft is commanded back to its default nadir-pointing attitude.

5.2.4. Pointing and Stabilization

Once a Cue satellite has selected a task and the required target boresight is known, the spacecraft attitude is commanded from its current Euler angles to the new target Euler angles. This maneuver is divided into two phases: a slew maneuver to rotate the spacecraft under actuator constraints, and a stabilization phase to suppress any residual oscillations before the observation can take place.

Slew Maneuver

The sensor retargeting maneuver is modelled through rotational dynamics, with constraints on the maximum angular velocity, $\omega_{\max}$, and angular acceleration, $\alpha_{\max}$. Therefore, the slew maneuver consists of three phases: an acceleration phase, where the angular velocity increases linearly at a rate of $\alpha_{\max}$ until it reaches $\omega_{\max}$, a constant-rate phase, where the rotation continues at a constant velocity of $\omega_{\max}$, and a deceleration phase, where the angular velocity decreases linearly to zero.

For small angular displacements, $\Delta\theta$, the maximum angular velocity $\omega_{\max}$ is never reached, and the maneuver simplifies to a triangular profile, consisting only of acceleration followed immediately by deceleration. The acceleration and deceleration times, t_{acc} and t_{dec}, are given by Equation 5.32.

$$t_{\text{acc}} = t_{\text{dec}} = \sqrt{\frac{\Delta\theta}{\alpha_{\max}}}, \qquad t_{\text{slew}} = 2\sqrt{\frac{\Delta\theta}{\alpha_{\max}}}. \tag{5.32}$$

For larger angular displacements that reach the maximum angular velocity, a trapezoidal profile is used. The time for the acceleration and deceleration phases is given by Equation 5.33.

$$t_{\text{acc}} = t_{\text{dec}} = \frac{\omega_{\max}}{\alpha_{\max}}, \qquad t_{\text{const}} = \frac{\Delta\theta - \frac{\omega_{\max}^2}{\alpha_{\max}}}{\omega_{\max}}, \tag{5.33}$$

Here, t_{acc} and t_{dec} are the acceleration and deceleration times, and t_{const} is the duration of the constant-rate phase. The total slew time can then be computed with Equation 5.34.

$$t_{\text{slew}} = t_{\text{acc}} + t_{\text{const}} + t_{\text{dec}}. \tag{5.34}$$

Assuming three independent reaction wheels, the total slew time for a 3D maneuver is taken as the maximum of the durations for each axis roll, pitch, and yaw.

Stabilization Phase

After the slew maneuver, the spacecraft must stabilize to suppress any residual oscillations. Stabilization is modeled as a second-order damped response with natural frequency ω_n and damping ratio ζ. The time required for the system to settle to 2% of the final value is approximated by Equation 5.35 [125].

$$t_{\text{settle}} \approx \frac{4}{\zeta\omega_n}. \tag{5.35}$$

To reflect that smaller rotations require less time to stabilize, the stabilization time is scaled with the magnitude of the commanded Euler-angle change and capped by the nominal settling time from Equation 5.35. The total time required for the maneuver is given by Equation 5.36.

$$t_{\text{point}} = t_{\text{slew}} + t_{\text{settle}}. \tag{5.36}$$

Time History and Execution in the Simulation

Once a task is selected and the slewing time is confirmed to be feasible, the attitude trajectory is generated at fixed timesteps. For each axis, the angular displacement is computed by integrating the corresponding triangular or trapezoidal slew profile. The resulting angular displacement is given in Equation 5.37, where τ represents the local time measured from the start of each phase.

$$\theta(\tau) = \begin{cases} \omega_0\tau + \frac{1}{2}\alpha_{\max}\tau^2, & \text{acceleration phase,} \\ \theta_{\text{acc}} + \omega_{\text{peak}}\tau, & \text{constant-rate phase,} \\ \theta_{\text{acc}} + \theta_{\text{const}} + \omega_{\text{peak}}\tau - \frac{1}{2}\alpha_{\max}\tau^2, & \text{deceleration phase,} \end{cases} \tag{5.37}$$

After the target attitude is reached, the attitude is held constant for the stabilization time, t_{settle}. During this time, the simulation applies prescribed residual angular velocities and accelerations before the final state is set to zero, preventing the observation from occurring before the spacecraft has fully stabilized.

This advance planning is required because the Cue satellite is typically tasked before the target enters its field-of-view (FoV). The spacecraft therefore slews immediately toward the predicted future observation attitude and completes both the slew and stabilization phases before the target enters the observable region, as illustrated in Figure 5.12.

Once the observation is complete, the task queue is checked again. If no other tasks are ready, the spacecraft is commanded back to its nadir-pointing attitude, with zero roll, pitch, and yaw, and it waits for new task requests from the Tip satellite.

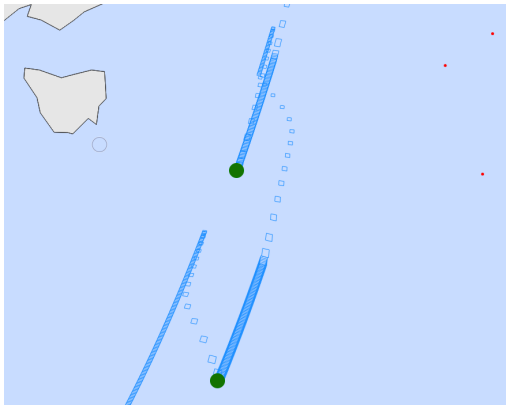


Figure 5.12: Field-of-view ground trace of the Cue satellite during tasking of two targets. After completing the first observation, the satellite slews back to nadir. Following the second acquisition, it returns to nadir again. The figure also illustrates how the satellite points ahead, completing the slewing maneuver with wider spaced field-of-view windows before reaching the target.

Data Logging and State Handling

To keep track of the target states during the simulation, five target states are logged, typically occurring in sequential order.

1. *Tip Observed*: The target falls within the field-of-view (FoV) of the Tip satellite.
2. *Tip Confirmed*: The onboard detection network of the Tip satellite has processed the target and sent a task request to the Cue satellite.
3. *Tasked*: This state occurs when the Cue satellite begins working on the task after completing other tasks.
4. *Cue Observed*: The target enters the FoV of the Cue satellite.
5. *Cue Confirmed*: The onboard detection system of the Cue satellite processes the target and confirms its status as either a whale or non-whale.

The confirmed targets of both Tip and Cue satellites are merged into two final lists: one for positive confirmations (whales) and one for negative confirmations (non-whales). In cases where the Tip satellite positively detects a target, the task is forwarded to the Cue satellite and added to the positive confirmed list once the Cue confirms it. Conversely, if the Tip satellite detects a target negatively, it is not forwarded to the Cue and instead is directly added to the negative confirmed list.

This results in five state lists that are used for both plotting and logging throughout the simulation: `observed_targets_tip`, `tasked_targets`, `observed_targets_cue`, `confirmed_targets_pos`, and `confirmed_targets_neg`.

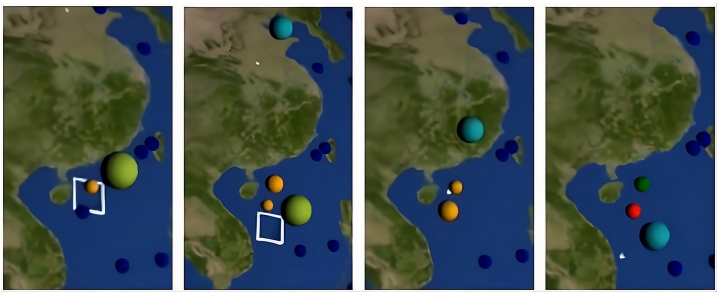


Figure 5.13: Simulation of Tip and Cue tasking logic, showing how a target is observed by the Tip satellite (orange), tasked (big orange), observed by the Cue satellite (small orange), and confirmed (true positive = green, false negative = red), with illumination and field-of-view conditions satisfied.

When the target falls within the viewing window of the Cue satellite, the following information is logged: target identifier, satellite coordinates (longitude, latitude, altitude), and the target's corresponding coordi-

nates (longitude, latitude, altitude), along with the observation time. This data is passed directly to the off-nadir rendering pipeline in section 5.3, and is required to later build the dataset used in the onboard detection decision-making process, as detailed in section 5.4.

There are two distinct methods for logging the off-nadir angle of the satellite. The first method calculates the off-nadir angle based on the satellite's actual Euler angles and the boresight reference frame. In this case, the off-nadir angle is computed relative to the satellite's actual boresight. However, when generating the off-nadir image, as described in section 5.3, the off-nadir angle is computed with respect to the line of sight to the target. This method does not account for the exact alignment between the satellite's boresight and the straight-line path to the target. Consequently, if the target is not positioned directly at the center of the image, the boresight of the satellite will not perfectly align with the line of sight to the target.

The order of the printed values in the image rendering pipeline deviates by a small fraction in the order of 0.05° with respect to the LVLH-computed value. This scale is to be expected and corresponds to a distance of:

$$0.05^\circ \times \frac{\pi}{180^\circ} \times R_{\text{earth}} = 5.56\,\text{km},$$

which closely matches the difference between a center point versus a border point observation, with a satellite ground track distance of 13.1 km. The actual boresight-target off-nadir angles are used in the satellite logging, corresponding to the actual satellite state.

Concluding the Orbital Simulation Framework, by propagating both satellite states and moving targets while accounting for illumination, field-of-view constraints, and attitude dynamics, the framework produces time-stamped observation geometries that define when and under which off-nadir conditions targets are observed.

These logged observation parameters form the input for the imaging stage of the framework. Specifically, the satellite position, target position, and time, from which the illumination conditions can be derived, are passed to the rendering pipeline, where the corresponding satellite images are generated. The following section therefore describes the off-nadir imaging approach used to simulate realistic observations under realistic illumination and viewing conditions.

5.3. Off-Nadir Imaging

The orbital simulation described in section 5.2 provides the satellite and target positions, viewing geometry, and solar illumination conditions for each observation event. To evaluate how these geometric conditions affect whale detectability, corresponding satellite imagery must be generated under varying off-nadir viewing angles.

This section describes the imaging pipeline used to synthesize these observations. Starting from nadir satellite images of whales, a physically-based rendering workflow is used to simulate off-nadir imaging conditions, including geometric distortion, illumination changes, and sun-glint effects. The pipeline combines scene geometry reconstruction, atmospheric radiative modelling, surface scattering models, and sensor spectral response modelling to produce radiance-consistent imagery that can be used for training and evaluating the onboard detection models.

5.3.1. Rendering Approach Selection

To evaluate the reflected radiance locally for each surface element under its specific illumination and viewing geometry, several radiance modelling approaches were first considered. These methods aim to numerically represent the Bidirectional Reflectance Distribution Function introduced in Figure 3.2.1, in order to estimate the radiance received by the sensor for a given viewing configuration. Because this function depends on multiple geometric, spectral, and surface-related variables, the available modelling approaches can be grouped into (i) radiative-transfer-based models, (ii) neural-network-based BRDF estimation methods, (iii) BRDF correction tools, and (iv) rendering engines.

Radiative Transfer-based Tools

Radiative-transfer-based tools, like Hydrolight, solve the scalar radiative transfer equation (RTE) for water bodies. They require the solar and sensor angles, wind speed (Cox–Munk slope statistics), inherent optical properties (absorption and scattering coefficients), water depth, and atmospheric conditions as input parameters. The output consists of BRDF tables, where the spectral radiance distributions are specified per wavelength and viewing direction, as visualized in Figure 5.14. While physically backed for ocean optics, the result is an angular radiance table rather than an image-level rendering. Integrating these outputs into a spatially varying off-nadir simulation would require pixel-wise lookup and additional geometric modelling. [126]

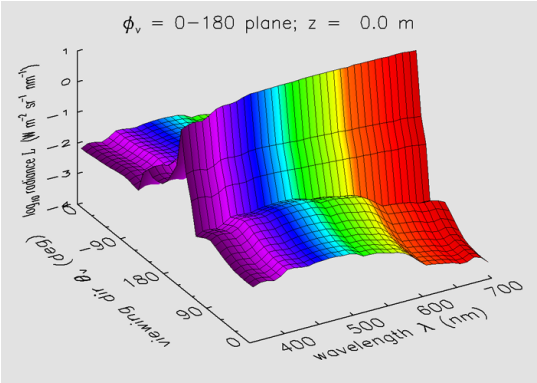


Figure 5.14: Example Hydrolight output showing spectral water-leaving radiance as a function of wavelength and viewing zenith angle in the principal plane ($\phi_v = 0°–180°$) at the water surface ($z = 0$ m). [1]

Neural Network-based Tools

Neural-network-based approaches, like BRDF-NeRF [48], estimate BRDF parameters and surface geometry directly from multi-view satellite images. The inputs consist of several RGB satellite images of the same scene acquired under different viewing and illumination angles, together with corresponding low-resolution depth maps obtained from stereo matching. The network predicts surface normals and RPV parameters of a semi-empirical BRDF model: amplitude, anisotropy, forward/backward scattering term, and hotspot term. These predicted parameters are then integrated into a differentiable renderer to synthesise views under different viewing conditions and recover a refined digital surface model. The outputs therefore include a synthetic images rendered under unseen viewing geometries, an estimated BRDF parameter maps, and a high-resolution depth model, as visualized in Figure 5.15. While this approach produces the desired output, it is originally trained on land surfaces. Expanding to marine applications would require multiple acquisitions of the same oceanic scene including a matching DEM oceanic surface model.

[1] Numerical Optics Ltd., "Hydrolight Radiative Transfer Software for Water." Available: `https://www.numericaloptics.com/hydrolight.html`. Accessed: Feb. 12, 2026.

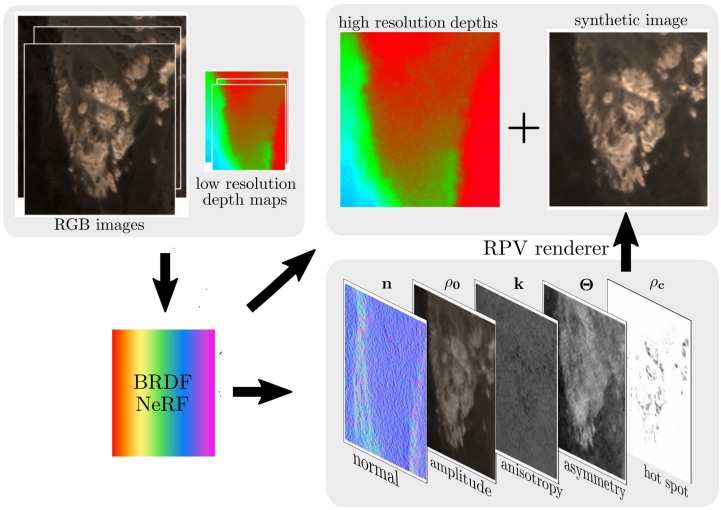


Figure 5.15: Overview of the BRDF-NeRF workflow. Multi-view satellite RGB images and their corresponding low-resolution depth maps are used as input. The network estimates surface normals and RPV BRDF parameters (ρ_0, k, Θ, ρ_c), describing amplitude, anisotropy, forward/backward scattering behaviour, and hotspot intensity. These parameters are integrated into an RPV-based renderer to synthesise radiance-consistent images under new viewing geometries. [48]

BRDF Correction Tools

Operational BRDF correction tools, like the Sentinel-3 OLCI BRDF correction processor (EUMETSAT) [127], are designed to remove directional effects by transforming directional water reflectance to a normalised reflectance referenced to a standard geometry, typically Sun at zenith and sensor at nadir, as visualized in Figure 5.16. This is implemented through a forward correction that applies a model-based normalisation factor, often derived from look-up tables and radiative transfer calculations, as a function of illumination/view geometry.

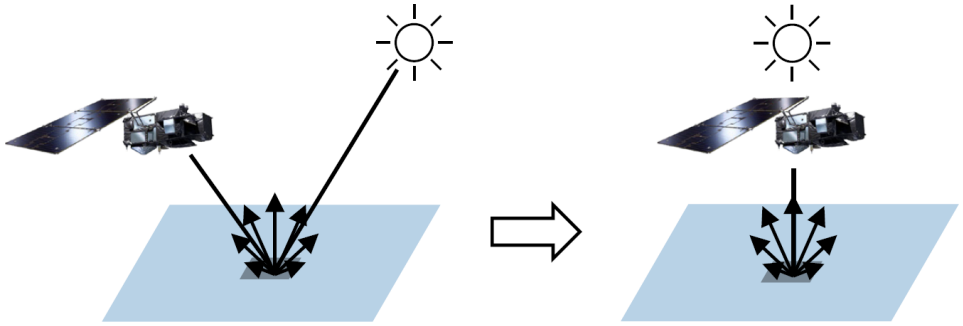


Figure 5.16: Illustration of BRDF correction geometry for water reflectance products. Left: measured (directional) reflectance under actual solar and sensor viewing geometry. Right: normalised reflectance after BRDF correction, referenced to a standard geometry with the Sun at zenith and the sensor at nadir. [127]

In principle, if the full forward model and all internal parameters were available, the inverse of the model could be applied to map reflectance from the reference geometry to another viewing configuration. However, most distributed products provide the normalised reflectance only. The intermediate directional components, lookup tables, and ancillary assumptions used during correction are not preserved in a form that allows reconstruction of the original directional radiance. Furthermore, the processing is based on the Sentinel-3 sensor geometry and spectral configuration, therefore, it cannot be used as a general forward radiance simulator for other satellites. Similar tools like Homonim [128], for which the output is visualized in Figure 5.17, perform BRDF normalisation and cross-sensor harmonisation, but also operate as correction frameworks rather than rendering models for forward simulation.

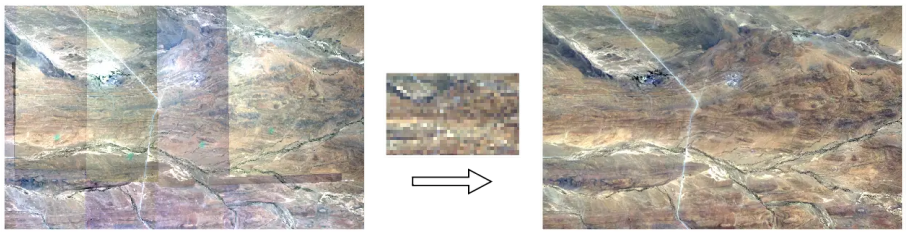


Figure 5.17: Example of BRDF normalisation and cross-sensor harmonisation using Homonim. Left: input satellite image with visible directional and radiometric inconsistencies. Right: corrected image after BRDF normalisation and harmonisation, showing radiometric consistency within the scene. [128]

Render-based Tools

Render-based approaches like Mitsuba [129] and PBRT [130] simulate light transport by numerically solving the rendering equation through Monte Carlo integration. In this framework, scene geometry, surface BRDF, illumination source, and sensor configuration are explicitly defined.

Mitsuba provides a modular Bidirectional Scattering Distribution Function (BSDF) system as illustrated in Figure 5.18, including microfacet models with Beckmann or GGX distributions, dielectric Fresnel terms, and spectral rendering capabilities. It should be noted that Mitsuba uses the term Bidirectional Scattering Distribution Function (BSDF) instead of the Bidirectional Reflectance Distribution Function (BRDF) introduced in Equation 3.1. The BSDF represents the sum of the reflected (BRDF) and transmitted (BTDF) components [131]. For the ocean surface considered here, reflection dominates. However the transmitted component is not fully negligible for the sensor geometry, because the Fresnel term redistributes incident energy between reflection and transmission, and the reflected glint magnitude depends on this coupling. Therefore, the BSDF formulation provides even an even more complete estimate of the outgoing radiation properties.

The workflow could consist of loading the captured image as a texture, defining the sun and satellite sensor location, assigning a BRDF model, and rendering the scene to obtain radiance-consistent imagery. Unlike Blender or Unity, which prioritize visual realism, Mitsuba and PBRT are designed for physically based light transport simulation. PBRT provides strong camera and lens modelling but requires C++ integration, whereas Mitsuba offers a Python interface and direct spectral rendering.

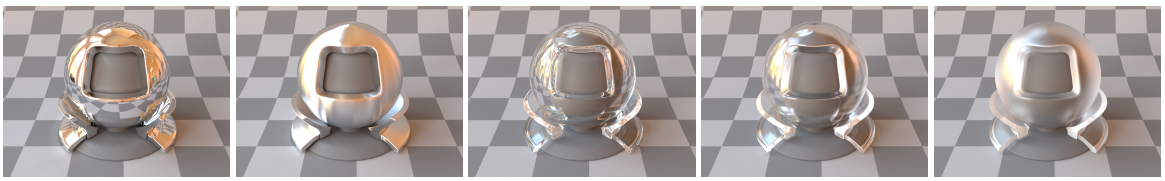


Figure 5.18: Comparison of Mitsuba Bidirectional Scattering Distribution Function (BSDF) plugins for conductors and dielectrics. From left to right: smooth conductor (`conductor`), rough conductor (`roughconductor`), smooth dielectric interface (`dielectric`), and rough dielectric interfaces (`roughdielectric`) with increasing microfacet roughness ($\alpha = 0.02$ and $\alpha = 0.1$). Conductors exhibit purely reflective Fresnel behavior (no transmission), while dielectrics both reflect and transmit. [129]

Considering the objectives of this thesis: forward simulation of off-nadir imagery including geometric distortion and sun glint, integration with orbital simulations, and controlled variation of viewing angles, a render-based approach is the most practical solution. Hydrolight provides accurate ocean optics but does not generate image-level scenes, and its output depends primarily on the oceanic parameters estimated. BRDF-NeRF is not optimized for dynamic water surfaces and would otherwise require multi-view oceanic training data and surface elevation models, which are not avaliable. Operational BRDF correction tools are designed for reflectance normalisation rather than forward modelling. Mitsuba, in contrast, allows explicit control over geometry, illumination, BRDF specification, and sensor configuration within a single physically based framework, while remaining practical to integrate into the AI-based Tip and Cue simulation pipeline.

For these reasons, Mitsuba was selected for the radiance modelling in this thesis.

Pipeline Overview
To introduce radiometric distortions to an existing dataset, the physically based rendering engine Mitsuba was used [129]. Mitsuba evaluates scene radiance by defining the illumination, sensor geometry, and surface scattering model. In this thesis, the spectral rendering variant is used for radiance generation, meaning that light transport is evaluated as a function of wavelength and subsequently integrated through the sensor spectral response, rather than relying on an RGB display transform. This is required to ensure that the simulated signal is consistent with the band-integrated radiance, which be converted to reflectance.

In the rendering workflow, an artificial DEM containing a wave-realistic ocean surface is first generated to provide the spatially varying surface normals required for the BRDF evaluation, after which the satellite position, target position, and Sun direction are converted into the local Mitsuba scene reference frame. Atmospheric illumination is then defined using a radiative transfer model to compute the spectral power distributions of direct solar irradiance and diffuse sky radiance, and a water surface BSDF is specified by combining a diffuse component representing the water-leaving radiance with a microfacet-based specular component to model sun glint.

Because the original satellite imagery is provided in RGB digital numbers, its pixel values do not directly represent physical reflectance. Therefore, the image is rescaled before being used as a diffuse texture in the renderer. Subsequently, satellite spectral response curves are defined so that each rendered band corresponds to a band-integrated radiance observation. The scene is then rendered to produce band-integrated at-sensor radiance values, after which the rendered spectral radiance is converted to reflectance to provide inputs compatible with the deep learning detection models. The geometric steps are described in subsection 5.3.2, while the radiative and sensor-modelling steps are described in subsection 5.3.3.

5.3.2. Scene Geometry Definition

Scene Set-up
To render a Mitsuba scene, all physically relevant entities have to be defined: the surface geometry, illumination sources, sensor model, sampling strategy, and light transport integrator. The complete scene definition is constructed as a dictionary and passed to Mitsuba's `load_dict` interface.

First, the target surface is defined. It consists of the artificially generated DEM mesh onto which the satellite image is projected through the selected BSDF:

```
        "earth_surface": {
            "type": "obj",
            "filename": dem_path,
            "to_world": to_world_scene,
            "bsdf": bsdf
        },
```

The DEM file generation and coordinate transformations are described in subsection 5.3.2, while the BSDF specification is introduced in subsection 5.3.3.

Illumination Model The total irradiance incident on the Earth's surface consists of two components: the direct solar irradiance arriving from a single direction, and the diffuse sky irradiance resulting from atmospheric scattering [132]. Consequently, two separate emitters are defined in the scene.

The Sun is modeled as a directional emitter:

```
        "sun": {
            "type": "directional",
            "direction": mi.ScalarVector3f(mi_light_dir),
            "irradiance": {
                "type": "spectrum",
                "filename": sun_spd
            }
        },
```

The direction vector `mi_light_dir` is computed relative to the target, and the spectral irradiance file `sun_spd` contains the direct normal spectral irradiance after atmospheric attenuation generated using SMARTS. These steps are described in subsection 5.3.2 and subsection 5.3.3 [133].

An alternative would be to model the Sun as a point source at its astronomical position. However, this would require explicit atmospheric modelling within the renderer to account for absorption and scattering before the radiation reaches the surface. By instead defining a directional emitter with irradiance already computed at ground level, atmospheric effects are consistently included without increasing computational complexity.

The diffuse sky component is defined as a constant environment emitter:

```
        "sky": {
            "type": "constant",
            "radiance": {
                "type": "spectrum",
                "filename": sky_spd
            }
        },
```

The constant emitter surrounds the scene and radiates isotropically toward it, representing hemispherical sky illumination. The spectral radiance distribution `sky_spd` is generated using SMARTS in subsection 5.3.3. Because this emitter also illuminates regions outside the DEM mesh, it produces a non-black background, which is postprocessed as described in subsection 5.3.4.

Sensor Definition The satellite sensor is defined as follows:

```
        "sensor": {
            "type": "perspective",
            "to_world": to_world_sensor,
            "fov": sensor_characteristics['fov_deg'],
            "far_clip": 1e8,
            "film": {
                "type": "specfilm",
                "width": sensor_characteristics['resolution'],
                "height": sensor_characteristics['resolution'],
                "spectral_band": {
                    "type": "spectrum",
                    "filename": spd_path
                }
            },
            "sampler": {
                "type": "independent",
                "sample_count": sensor_characteristics['sample_count']
            }
        }
```

Mitsuba supports multiple camera models. An `orthographic` sensor generates parallel rays and does not model perspective effects. A `perspective` sensor represents a pinhole camera with a defined field of view. A `thin-lens` model extends the perspective camera with finite aperture size and depth-of-field simulation. Since satellite instruments are assumed to be radiometrically and geometrically calibrated and depth-of-field effects are negligible at orbital altitudes, a `perspective` camera model is selected.

The field of view and the transformation matrix `to_world_sensor` are computed from the scene geometry described in subsection 5.3.2. The parameter `far_clip` defines the maximum rendering distance and is set to 10^8 m (10,000 km), which is sufficient for Low Earth Orbit configurations while limiting unnecessary ray traversal.

To preserve spectral information, a film of type `specfilm` is used instead of `hdrfilm`. The `specfilm` integrates spectral radiance using the satellite-specific spectral response curve $R_i(\lambda)$ provided in `spd_path`,

as described in subsection 5.3.3. The band-integrated radiance recorded by the sensor is therefore given by

$$L_i = \int L(\lambda)\, R_i(\lambda)\, d\lambda, \tag{5.38}$$

where $L(\lambda)$ is the spectral radiance computed by the renderer. This ensures that the output radiance per band is physically consistent with the sensor characteristics.

The sampler is set to `independent` with a predefined sample count, as discussed in subsection 5.3.4. Finally, light transport is simulated using a path tracing integrator:

```
        "type": "scene",
        "integrator": {"type": "path"},
```

The path integrator numerically evaluates the rendering equation by Monte Carlo integration, accounting for multiple scattering events between emitters and surfaces. The implications of integrator and sampling choices on variance and convergence are discussed in subsection 5.3.4.

Artificial DEM File Generation
Because the BSDF evaluation depends on the local incident and reflected zenith angles, the ocean surface geometry must be represented. Given an input satellite image of resolution $N \times N$, an artificial elevation field is generated at matching spatial resolution, and converted into a triangulated mesh that can be rendered in Mitsuba. This ensures that spatially varying surface normals are present for the BSDF to be evaluated.

Wave Field Synthesis The sea surface elevation is modeled as a finite superposition of harmonic wave components as given in Equation 5.39 [134].

$$\eta(x, y, t) = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{i,j} \cos\left(\omega_i t - k_i x \cos\theta_j - k_i y \sin\theta_j + \alpha_{i,j}\right). \tag{5.39}$$

Because a static surface snapshot is sufficient for rendering, t is fixed and the temporal term $\omega_i t$ is absorbed into the random phase $\alpha_{i,j}$. The discrete elevation field therefore simplifies to Equation 5.40.

$$\eta(x, y) = \sum_{n=1}^{N_w} a_n \cos\left(k_{x,n} x + k_{y,n} y + \phi_n\right), \tag{5.40}$$

where N_w is the number of sampled wave components, and harmonic constants $k_{x,n}$ and $k_{y,n}$ given in Equation 5.41.

$$k_{x,n} = \frac{2\pi}{\lambda_n} \cos\theta_n, \qquad k_{y,n} = \frac{2\pi}{\lambda_n} \sin\theta_n. \tag{5.41}$$

Here, λ_n is the sampled wavelength, θ_n the propagation direction, a_n the amplitude, and ϕ_n a random phase between 0 and 2π. The wavelengths, amplitudes, directions, and phases are randomly sampled within predefined bounds.

The resulting elevation field is normalized and rescaled to the interval `[wave_min, wave_max]` to control the height range and resulting slope distribution. Although this does not fully reproduce a specific ocean spectrum, it generates a realistic local slope variability driving the angular response of the BSDF.

Earth Curvature Approximation To ensure geometric consistency with large image footprints, curvature of the Earth is added. The curvature is modeled as the height of an arc of a sphere, also referred to as the sagitta, as shown in Figure 5.19. With radius R evaluated at radial distance $D(x, y) = \sqrt{x^2 + y^2}$, the the sagitta is given by Equation 5.42 [135].

$$S(x, y) = R - \sqrt{R^2 - D(x, y)^2}. \tag{5.42}$$

Combining Equation 5.40 and Equation 5.42, the final elevation field can then be computed with Equation 5.43.

$$Z(x, y) = \eta(x, y) - S(x, y), \tag{5.43}$$

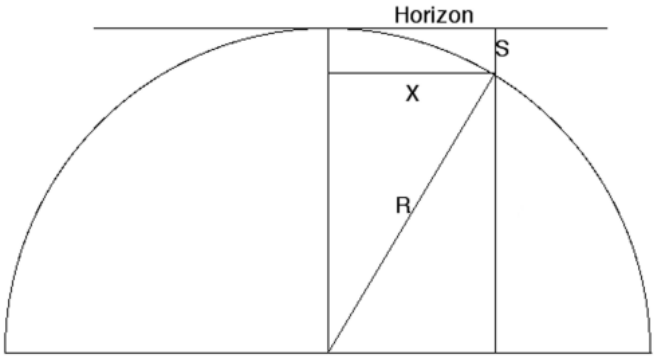


Figure 5.19: Geometry of Earth curvature. The circle represents the Earth with radius R, the horizontal line is the local horizon (tangent). At surface point X from the tangent point, the sagitta due to curvature is given by S. [135]

Mesh Construction The elevation raster $Z(x, y)$ is stored as a single-band GeoTIFF and then converted to a triangulated mesh. Given image resolution (W, H) and ground sampling distance GSD, the physical image dimensions are given by Equation 5.44.

$$W_{\mathrm{m}} = W \cdot \mathrm{GSD}, \qquad H_{\mathrm{m}} = H \cdot \mathrm{GSD}. \tag{5.44}$$

A centered Cartesian grid is constructed over these extents and vertices are defined as $\mathbf{v}_{i,j} = \big(x_j,\, y_i,\, Z(x_j, y_i)\big)$.

Each grid cell is subdivided into two triangles to form a continuous surface. Linear UV coordinates are assigned over $[0, 1]^2$ to ensure one-to-one mapping between DEM vertices and image pixels. The resulting OBJ mesh can then be loaded into Mitsuba.

The generated elevation field and corresponding triangulated surface are visualized in Figure 5.20.

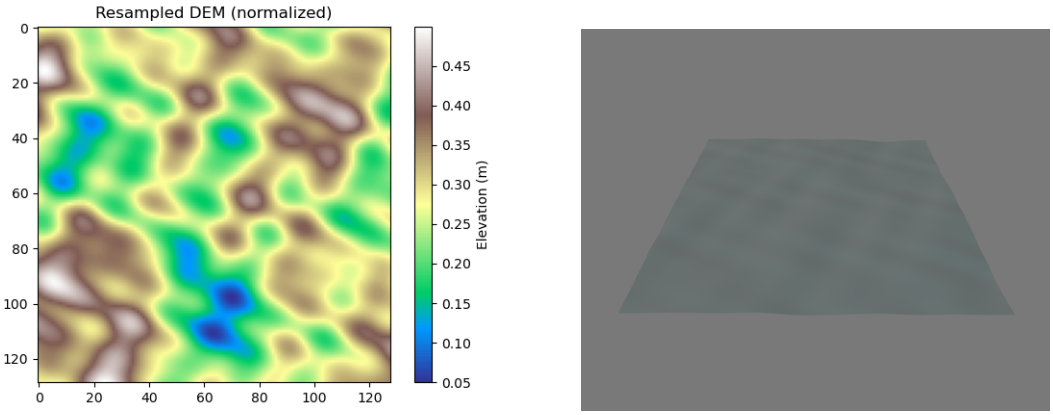


Figure 5.20: Generated Digital Elevation Models (DEM) used to represent the ocean surface geometry. The left image shows the normalized wave height field visualized as a contour plot. The right image presents the corresponding three-dimensional visualization of the generated DEM surface used within the rendering pipeline.

Coordinate Transformations

After generating the DEM, Mitsuba defines the scene in a local Cartesian reference frame. Therefore the target, satellite, and Sun vectors computed in Earth-Centered Earth-Fixed (ECEF) coordinates must be expressed relative to the target location.

Local reference frame Let $\mathbf{p}_{\text{tgt}}^{\text{ecef}}$, $\mathbf{p}_{\text{sat}}^{\text{ecef}}$, and $\mathbf{p}_{\odot}^{\text{ecef}}$ denote the ECEF position vectors of target, satellite, and Sun, obtained from latitude and longitude.

A local target-centered frame is defined with its origin at the target, with axes alignment following the ECEF axes vectors. The local coordinates are obtained by subtracting the target position and applying a rotation from ECEF to a local tangent frame as shown in Equation 5.45.

$$\mathbf{p}^{\text{loc}} = \mathbf{R}_{\text{loc}\leftarrow\text{ecef}} \left(\mathbf{p}^{\text{ecef}} - \mathbf{p}_{\text{tgt}}^{\text{ecef}} \right). \tag{5.45}$$

Here, the local frame is chosen so that the z-axis points upward at the target and Mitsuba can use `up=[0,0,1]` us upward direction. A practical choice for $\mathbf{R}_{\text{loc}\leftarrow\text{ecef}}$ is the East-North-Up (ENU) reference frame centered at the target geodetic latitude φ and longitude λ, with its rotation matrix given in Equation 5.46.

$$\mathbf{R}(\varphi, \lambda) = \begin{bmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ \cos\varphi\cos\lambda & \cos\varphi\sin\lambda & \sin\varphi \end{bmatrix}. \tag{5.46}$$

Sun direction Mitsuba requires a direction vector for the directional Sun emitter. The incoming ray direction at the target is the unit vector from the target towards the Sun $\hat{\boldsymbol{\omega}}_{\odot}^{\text{loc}}$, which can be computed with Equation 5.47.

$$\hat{\boldsymbol{\omega}}_{\odot}^{\text{loc}} = \frac{\mathbf{R}_{\text{loc}\leftarrow\text{ecef}} \left(\mathbf{p}_{\odot}^{\text{ecef}} - \mathbf{p}_{\text{tgt}}^{\text{ecef}} \right)}{\left\| \mathbf{R}_{\text{loc}\leftarrow\text{ecef}} \left(\mathbf{p}_{\odot}^{\text{ecef}} - \mathbf{p}_{\text{tgt}}^{\text{ecef}} \right) \right\|}. \tag{5.47}$$

Off-nadir angle and azimuth The viewing direction from the target to the satellite in the local frame is given in Equation 5.48.

$$\hat{\boldsymbol{\omega}}_{\text{view}}^{\text{loc}} = \frac{\mathbf{p}_{\text{sat}}^{\text{loc}}}{\left\| \mathbf{p}_{\text{sat}}^{\text{loc}} \right\|}. \tag{5.48}$$

The geocentric off-nadir and azimuth angles, used for reporting and for rotating the DEM, are then computed with respect to the local up-axis $\mathbf{k} = [0, 0, 1]^\top$ with Equation 5.49 and Equation 5.50 respectively.

$$\theta_{\text{off}} = \arccos\left(\hat{\boldsymbol{\omega}}^{\text{loc}}_{\text{view}} \cdot \mathbf{k}\right), \tag{5.49}$$

$$\phi_{\text{az}} = \operatorname{atan2}\left(\hat{\boldsymbol{\omega}}^{\text{loc}}_{\text{view},y}, \hat{\boldsymbol{\omega}}^{\text{loc}}_{\text{view},x}\right). \tag{5.50}$$

ϕ_{az} is used to rotate the surface mesh around the local z-axis. The applied transform is the composition of a rotation and a mirroring operation from Equation 5.51, to ensure that the the textured DEM orientation aligns with the original image coordinates.

$$\mathbf{T}_{\text{scene}} = \mathbf{R}_z(-\phi_{\text{az}})\,\mathbf{M}_x, \qquad \mathbf{M}_x = \operatorname{diag}(-1,\ 1,\ 1), \tag{5.51}$$

Field of view Given an image of $N \times N$ resolution and a ground sampling distance (GSD) in meters per pixel, the ground-projected patch width is $W = N \cdot \text{GSD}$, following Equation 5.44. With sensor distance $r = \left\|\mathbf{p}^{\text{ecef}}_{\text{sat}} - \mathbf{p}^{\text{ecef}}_{\text{tgt}}\right\|$, the camera field-of-view used by Mitsuba can be approximated with Equation 5.52, which is passed on to the perspective sensor.

$$\text{FOV} = 2\,\arctan\left(\frac{W/2}{r}\right), \tag{5.52}$$

5.3.3. Radiative and Sensor Modelling

Radiative Transfer Model

To compute the spectral power distributions required for scene illumination, SMARTS is used as an atmospheric radiative transfer model [136], called through the Python wrapper pySMARTS [137]. The model is evaluated at the target location and acquisition time, using as inputs the target geodetic position (latitude, longitude, altitude), the UTC timestamp, a wavelength range of 250-1000 nm, and the surface material set to `Water`. The selected wavelength interval covers the visible domain and provides margin for verification on a broader spectrum.

Two SMARTS outputs are extracted. The first is the direct normal spectral irradiance $E_{\text{dir,n}}(\lambda)$ (SMARTS: `Direct normal irradiance W m-2`), which is used to define the solar spectral power distribution for the directional emitter. The second is the diffuse horizontal spectral irradiance $E_{\text{dif,h}}(\lambda)$ (SMARTS: `Diffuse horizontal irradiance W m-2`), which is used to define the sky illumination.

Since Mitsuba's constant environment emitter expects a spectral radiance distribution, the diffuse horizontal irradiance is converted to an equivalent isotropic sky radiance by assuming a Lambertian sky dome. Under this assumption, the hemispherical irradiance is related to radiance by a factor π, so the sky spectral radiance is computed as in Equation 5.53.

$$L_{\text{sky}}(\lambda) = \frac{E_{\text{dif,h}}(\lambda)}{\pi}. \tag{5.53}$$

Both spectra are written to disk as two-column spectral power distribution (SPD) files compatible with Mitsuba's `spectrum` file input, using wavelength in nm and spectral values in SI units.

The resulting files are then used directly in the scene definition to parameterize the solar irradiance spectrum for the directional emitter and the sky radiance spectrum for the constant environment emitter.

An example result of both sun and sky spectral response, for a given time and target position, is shown in Figure 5.21.

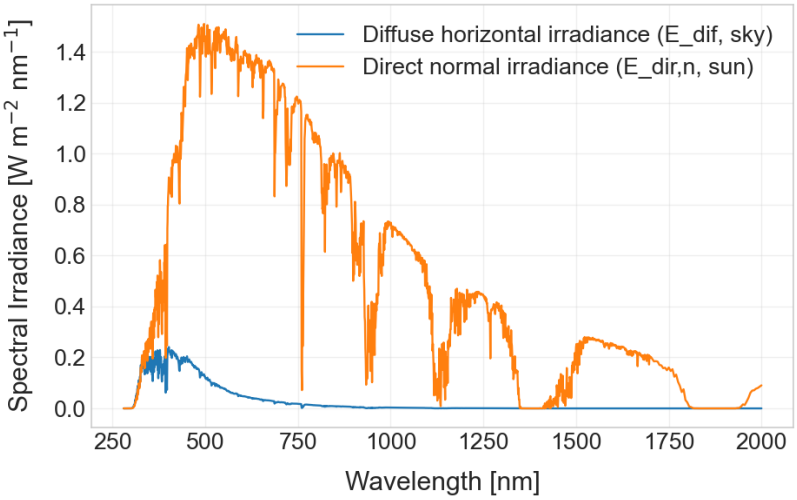


Figure 5.21: Example SMARTS-derived atmospheric spectra used to parameterize Mitsuba illumination: direct normal spectral irradiance at the surface for the directional Sun emitter, and diffuse horizontal spectral irradiance for the sky component. The wavelength-dependent attenuation and absorption are visible as narrow dips in the direct component, while the diffuse component is smoother and lower in magnitude.

Bidirectional Scattering Distribution Function

Mitsuba provides multiple Bidirectional Scattering Distribution Functions (BSDFs), like surface scattering models that describe how incident radiance is redistributed into reflected and/or transmitted directions. An overview of the material families is shown in Figure 5.22. In short, `diffuse` models ideal Lambertian reflection, `dielectric` models a smooth refractive interface with both reflection and transmission, `thindielectric` approximates a thin refractive sheet without angular deflection in transmission, `conductor` models a purely reflective metallic interface, and `plastic` combines diffuse and specular contributions through an internal coating model. For several of these families, rough variants (`roughdielectric`, `roughconductor`, `roughplastic`) replace the ideal specular interface by a microfacet distribution, producing a finite-width specular lobe instead of a Dirac-delta reflection [129].

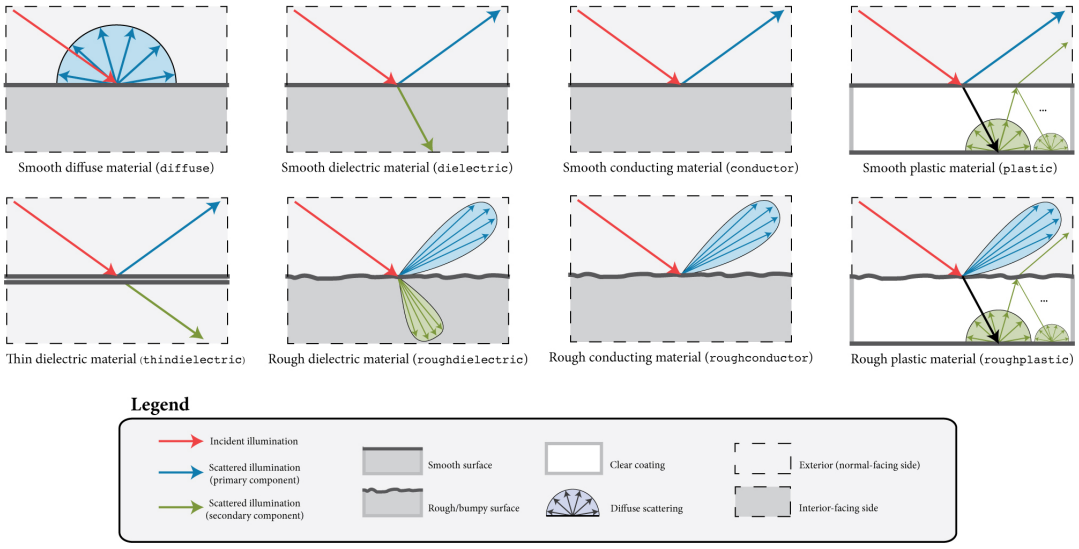


Figure 5.22: Overview of the principal BSDF models available in Mitsuba. The top row shows smooth material models (diffuse, dielectric, conductor, and plastic), where specular reflection is represented as an ideal, perfectly smooth interface. The bottom row presents the corresponding rough variants, in which surface microgeometry is modeled through a microfacet distribution, producing an angular distribution of reflected energy rather than a single specular direction. [129]

When selecting a BSDF for this use case, the microfacet formulation in Equation 3.6 can approximate the

model basis, and the conceptual decomposition into diffuse, mirror, and glossy contributions is illustrated in Figure 5.23. In the satellite rendering pipeline, the loaded image is used as a diffuse reflectance texture to represent the whale with background water appearance (Lambertian approximation), while the glossy sun-glint contribution is modelled by a rough microfacet interface through `roughdielectric`, containing both the microfacet distributions for reflection and transmission.

The ideal mirror term is implicitly approached by the microfacet model when the roughness approaches zero. In practice, however, a perfectly smooth surface with zero roughness, for which all reflected energy is concentrated into a single direction, is highly unlikely for this use case. The presence of whales further disrupts the continuity of the water surface. Together with wave-induced roughness, this is also consistent with visual observation. One may see oneself reflected in the surface of a calm lake, but not in the ocean, where the higher surface roughness suppresses a strong mirror component. Therefore, for the present use case, the BSDF response is dominated by the diffuse and glossy components.

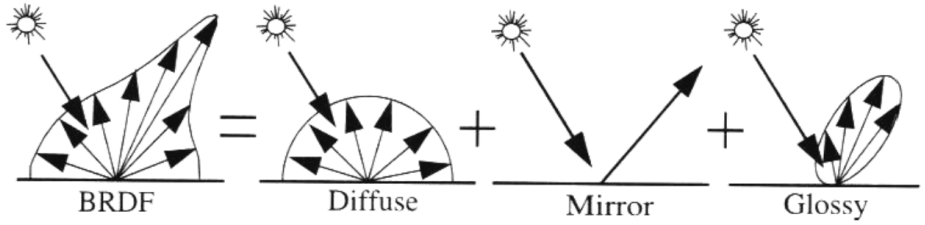


Figure 5.23: Decomposition of the Bidirectional Reflectance Distribution Function (BRDF) into its reflection components. The total BRDF response can be interpreted as the sum of diffuse reflection (Lambertian scattering), ideal specular reflection (mirror component), and glossy reflection representing microfacet-based specular scattering with finite roughness. [2]

Diffuse texture component The input image is mapped onto the ocean mesh as a diffuse BSDF with a bitmap texture serving as its reflectance parameter. Because `diffuse` expects reflectance-like values as input, the RGB image values are rescaled before being provided as texture input, as described in subsection 5.3.3.

Rough dielectric sun-glint component To model glint, the scripts select Mitsuba's `roughdielectric` BSDF, configured with a Beckmann microfacet distribution and indices of refraction set to water for the interior (1.3330) and air for the exterior (1.000277) [129].

The roughness parameter α is computed from an input wind speed V_{wind} using the Cox-Munk slope statistics, which relation is given in Equation 5.54. [138]

$$\alpha = \sqrt{0.003 + 0.00512\, V_{\text{wind}}}. \tag{5.54}$$

Blending diffuse and glint After defining both components, Mitsuba's `blendbsdf` is used to linearly mix the diffuse texture BSDF and the rough dielectric BSDF using a scalar specular weight $w \in [0, 1]$. The resulting BSDF is the combination in Equation 5.55.

$$f_{\text{blend}} = (1 - w)\, f_{\text{diffuse}} + w\, f_{\text{roughdielectric}}. \tag{5.55}$$

Here, w corresponds to the `specular_weight` parameter in the scene definition, where $w = 0$ gives a purely diffuse appearance and $w = 1$ gives a purely rough dielectric interface.

[3] M. Vlnas, "Light Transport Simulation: From Basics to Advanced," Brno University of Technology, Faculty of Information Technology, Oct. 31, 2022. Available: `https://www.fit.vut.cz/person/ivlnas/public/shared/pgp-light-transport.pdf`. Accessed: Feb. 13, 2026.

RGB to Reflectance Estimation

Mitsuba's `diffuse` BSDF expects a (spectral or RGB) reflectance/albedo input. The available dataset, however, provides only an 8-bit RGB image with unknown radiometric calibration and an unknown mapping between digital numbers and surface reflectance. Therefore, an approximate reflectance proxy has to be constructed.

The water region is first identified using the whale masks from COCO annotations, so that the whale pixels can be excluded, as shown in Figure 5.24. Assuming that the anchor region is representative open water, the per-channel response is monotonic, and each RGB channel can be scaled independently, the image is converted to linear intensity and then rescaled so that the median water intensity matches a target water reflectance vector.

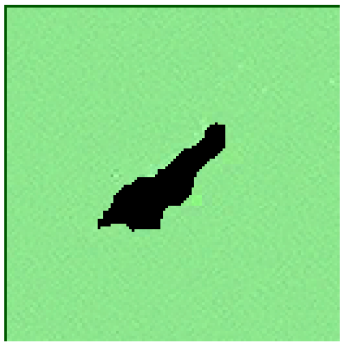


Figure 5.24: Water-region mask derived from the COCO whale annotation. Whale pixels are excluded (black), and the remaining pixels define the open-water reference region (green) used to compute per-channel median intensities for RGB-to-reflectance rescaling.*Satellite image © 2022 Maxar Technologies.*

The RGB image is first converted from 8-bit digital numbers to linear intensity by applying an inverse gamma correction with $\gamma = 2.2$. The water mask is then used to estimate, for each channel independently, a scale factor so that the median linear intensity over the water pixels matches a target water reflectance. Finally, the full image is multiplied by these per-channel scale factors and clipped to $[0, 1]$ to obtain a reflectance proxy that can be used as input to Mitsuba's `diffuse` BSDF.

However, target water reflectance is not known beforehand. In this work, it is estimated by rendering a fully glint-dominated reference case, with the specular weight set to 1.0, under the same solar and viewing geometry. The resulting band-integrated at-sensor radiance is converted to reflectance, and channel statistics (median over the masked footprint) are computed. These median values define the target reflectance vector.

This approach assumes that, for a fixed illumination and viewing geometry, the overall reflectance level of open water is primarily governed by Fresnel reflection and surface slope statistics, which are explicitly modeled in the microfacet term. Under identical geometry, the rendered glint-only case therefore provides a physically consistent estimate of the reflectance magnitude associated with the water surface. Because both the rendered reference and the rescaled image correspond to the same solar zenith angle, viewing direction, and surface roughness parameter. the RGB image is then scaled so that its median water response matches this physically evaluated reference level. The enforces consistency in reflectance magnitude while preserving the spatial structure of the observed image.

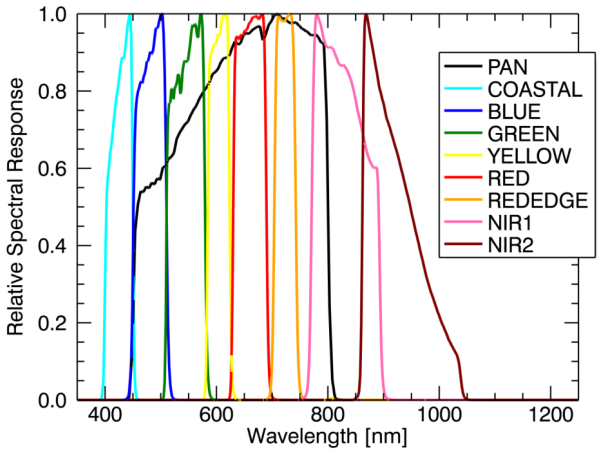
Nevertheless, this remains an approximation: it does not recover absolute reflectance calibration, it assumes linear per-channel scaling, and it propagates unknown sensor processing steps like white balance, tone mapping, and compression into the fitted scale factors. However, given the absence of sensor calibration metadata, this allows the diffuse BSDF to represent water-leaving radiance patterns while the physically evaluated microfacet term models sun glint separately, while preserving a geometry-consistent reflectance magnitude.

Satellite Spectral Response Curves

The satellite spectral response curves describe how the sensor weights incoming radiance as a function of wavelength for each spectral band. They define the bandpass function $R_i(\lambda)$ from Equation 5.38, which

determines how spectral radiance is integrated into discrete band measurements.

Assuming a WorldView-3-like satellite, these response functions can be obtained from the radiometric use documentation provided by Maxar [139]. The documentation provides the relative spectral response plots together with tabulated center wavelengths λ_c and effective bandwidths $\Delta\lambda$, as shown in Figure 5.25.



Spectral Band	Center Wavelength (nm)	Effective Bandwidth, $\Delta\lambda$ (µm)
Panchromatic	649.4	0.2896
Coastal	427.4	0.0405
Blue	481.9	0.0540
Green	547.1	0.0618
Yellow	604.3	0.0381
Red	660.1	0.0585
Red Edge	722.7	0.0387
NIR1	824.0	0.1004
NIR2	913.6	0.0889
SWIR1	1209.1	0.0330
SWIR2	1571.6	0.0397
SWIR3	1661.1	0.0373
SWIR4	1729.5	0.0416
SWIR5	2163.7	0.0389
SWIR6	2202.2	0.0409
SWIR7	2259.3	0.0476
SWIR8	2329.2	0.0679

Figure 5.25: Spectral characteristics of the WorldView-3 sensor. The left image shows the relative spectral response functions of the spectral bands. The right table lists the corresponding center wavelengths and effective bandwidths. [139]

Since the manufacturer provides the response curves only graphically and through tabulated band parameters, the spectral response functions must be reconstructed numerically and exported as script-readable spectral power distribution (SPD) files for use within Mitsuba.

For each band i, defined by its lower and upper wavelength limits λ_L and λ_U, the spectral response is approximated by a smooth Gaussian error function $\mathrm{erfc}(\cdot)$ band-pass given in Equation 5.56, producing a passband between λ_L and λ_U with smooth transitions at the band edges [140].

$$R_i(\lambda) = H_{\mathrm{HP}}(\lambda)\, H_{\mathrm{LP}}(\lambda), \tag{5.56}$$

Here, β controls the selectivity and f_t is the transition midpoint. A band-pass response is obtained by combining a high-pass and low-pass response given in Equation 5.57

$$\begin{aligned} H_{\mathrm{HP}}(\lambda) &= 1 - 0.5\ \mathrm{erfc}\big(2\beta(\lambda - \lambda_L)\big)\,, \\ H_{\mathrm{LP}}(\lambda) &= 0.5\ \mathrm{erfc}\big(2\beta(\lambda - \lambda_U)\big)\,. \end{aligned} \tag{5.57}$$

Here, λ_L and λ_U represent the lower and upper band limits, which are derived from the the center wavelength λ_c from Figure 5.25.

$$\lambda_L = \lambda_c - \frac{\Delta\lambda}{2}, \qquad \lambda_U = \lambda_c + \frac{\Delta\lambda}{2}. \tag{5.58}$$

The parameter β determines the transition steepness and is computed with Equation 5.59

$$\beta = \frac{\mathrm{erfc}^{-1}(2\varepsilon)}{\Delta\lambda_{\mathrm{tr}}}, \tag{5.59}$$

where ε defines the stopband attenuation and $\Delta\lambda_{\mathrm{tr}}$ is the transition width. In this work, the overlap level is chosen as $\delta = 1/\sqrt{2}$ (-3.01 dB), consistent with the formulation in [140].

After evaluation, each response is normalized so that $\max_\lambda R_i(\lambda) = 1$.

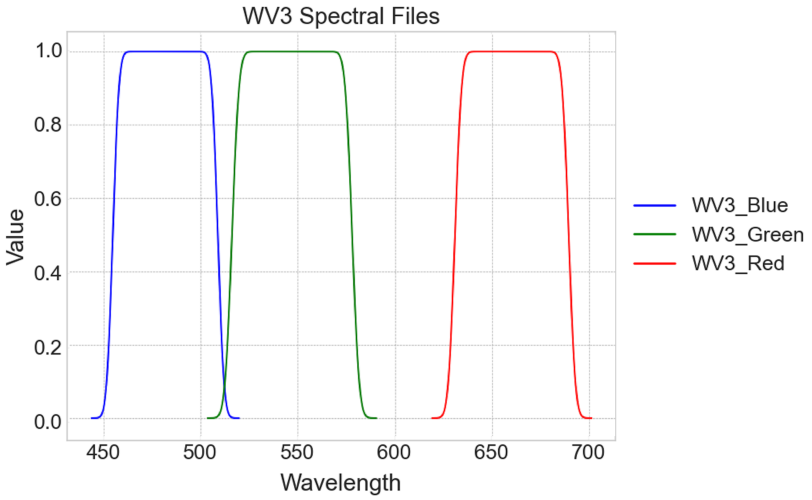


Figure 5.26: Generated spectral response functions for the blue, green, and red bands of the WorldView-3 sensor using the erfc-based band-pass formulation of [140].

The result of the erfc-based approach is shown in Figure 5.26. Compared to the Maxar relative spectral response curves for WV3, the reconstructed passbands are still too idealized: within the nominal bandwidth the response is nearly flat, while the published WorldView-3 curves show a in-band rise and additional fine-scale structure [139].

The effect is that the band-integrated radiance will be overestimated, which deminishes the physical accuracy of the results. Since the measured band signal is computed as $L_i = \int_{\lambda_L}^{\lambda_U} L(\lambda)\, R_i(\lambda)\, d\lambda$, a flat-top approximation increases the effective area $A_i = \int_{\lambda_L}^{\lambda_U} R_i(\lambda)\, d\lambda$ compared to a rising in-band response. For spectra with non-uniform spectral structure $L(\lambda)$, this leads to a biased weighting and therefore physically inconsistent radiance levels, even if the nominal bandwidth $\Delta\lambda$ is preserved.

To reduce this bias while retaining the erfc edge transitions from Equation 5.56–Equation 5.57, a in-band slope is introduced. A normalized spectral coordinate is defined as in Equation 5.60.

$$t(\lambda) = \mathrm{clip}\left(\frac{\lambda - \lambda_L}{\lambda_U - \lambda_L}, 0, 1\right), \tag{5.60}$$

This maps the lower band edge λ_L to 0 and the upper band edge λ_U to 1. This dimensionless coordinate represents the relative position within the passband and acts as a slope parameter.

The tilt function is then defined as in Equation 5.61.

$$T_i(\lambda) = \eta + (1 - \eta)\, t(\lambda)^p, \tag{5.61}$$

Here, η determines the relative response level at λ_L and p controls the curvature of the rise toward λ_U. The final shaped response is obtained by blending this slope into the erfc band-pass, followed by renormalization to $\max_\lambda R_i(\lambda) = 1$. The result is shown in Figure 5.27. Comparing to Figure 5.25, it can be observed that this modification preserves the bandwidth and incline, which resembles the most important features of the WV3 spectral power distribution curves.

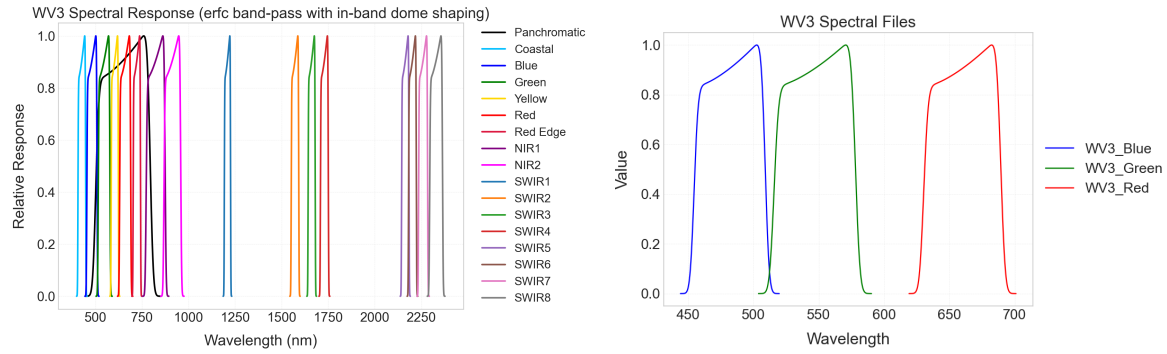


Figure 5.27: Generated spectral response functions for spectral bands of the WorldView-3 sensor using the erfc-based band-pass formulation of [140] and tilt function, with the full spectrum on the left, and RGB bands isolated on the right.

Spectral Radiance to Reflectance Conversion

After rendering, the sensor output captures the band-integrated radiance per spectral band as defined in Equation 5.38. To match the reflectance representation used by the downstream Deep Learning models, the rendered radiance is converted to top-of-atmosphere (TOA) reflectance.

The TOA reflectance $\rho_{\text{TOA,band}}$ is computed independently for the red, green, and blue bands using Equation 5.62 [139].

$$\rho_{\text{TOA,band}} = \frac{\pi \bar{L}_{\lambda,\text{band}} d^2}{\bar{E}_{\text{band}} \cos\theta_S}, \tag{5.62}$$

where d is the Earth-Sun distance in astronomical units (AU), θ_S is the solar zenith angle with respect to the target surface, $\bar{E}_{\text{band}}$ is the band-averaged solar irradiance, and $\bar{L}_{\lambda,\text{band}}$ is the band-averaged spectral radiance at the sensor.

The solar zenith angle θ_S is derived from the Earth-centered geometry. Let $\mathbf{r}_T$ denote the target position vector in ECEF coordinates and $\mathbf{r}_\odot$ the Sun position vector in the same frame. The local surface normal, assuming a spherical Earth, is defined in Equation 5.63.

$$\mathbf{n} = \frac{\mathbf{r}_T}{\|\mathbf{r}_T\|}, \tag{5.63}$$

The direction from the target toward the Sun is defined in Equation 5.64.

$$\mathbf{s} = \frac{\mathbf{r}_\odot - \mathbf{r}_T}{\|\mathbf{r}_\odot - \mathbf{r}_T\|}, \tag{5.64}$$

The cosine solar zenith term is then computed as the dot product in Equation 5.65.

$$\cos\theta_S = \mathbf{n} \cdot \mathbf{s}. \tag{5.65}$$

Next, the band-averaged solar irradiance in Equation 5.62 is defined in Equation 5.66.

$$\bar{E}_{\text{band}} = \frac{\int E_\odot(\lambda)\, R(\lambda)\, d\lambda}{\int R(\lambda)\, d\lambda}. \tag{5.66}$$

The numerator of Equation 5.66 corresponds to the response-weighted in-band solar irradiance, defined in Equation 5.67.

$$E_{\text{band}} = \int E_\odot(\lambda)\, R(\lambda)\, d\lambda, \tag{5.67}$$

The denominator of Equation 5.66 corresponds to the effective bandwidth (area under the response curve), defined in Equation 5.68.

$$\Delta\lambda = \int R(\lambda)\, d\lambda. \tag{5.68}$$

The integrals in Equation 5.67 and Equation 5.68 are evaluated directly from the spectral power distribution (SPD) files. The solar SPD files that describes $E_{\odot}(\lambda)$ provides tabulated $(\lambda_k, E_{\odot,k})$ pairs. Similarly, the band response SPD file represents $R(\lambda)$ and provides (λ_j, R_j) pairs.

Both spectra are then resampled onto a common wavelength grid and the integrals in Equation 5.67 and Equation 5.68 are approximated using the trapezoidal rule as given in Equation 5.69 and Equation 5.70.

$$E_{\text{band}} \approx \sum_{i=1}^{N-1} \frac{(E_{\odot,i} R_i + E_{\odot,i+1} R_{i+1})}{2} (\lambda_{i+1} - \lambda_i), \tag{5.69}$$

$$\Delta\lambda \approx \sum_{i=1}^{N-1} \frac{(R_i + R_{i+1})}{2} (\lambda_{i+1} - \lambda_i). \tag{5.70}$$

The rendered spectral integrator outputs the band-integrated radiance L_{band} defined in Equation 5.71.

$$L_{\text{band}} = \int L(\lambda)\, R(\lambda)\, d\lambda, \tag{5.71}$$

To obtain the band-averaged spectral radiance required in Equation 5.62, the radiance is normalized by the effective bandwidth from Equation 5.68, as given in Equation 5.72.

$$\bar{L}_{\lambda,\text{band}} = \frac{L_{\text{band}}}{\Delta\lambda}. \tag{5.72}$$

Substituting Equation 5.72, Equation 5.66, and Equation 5.65 into Equation 5.62 gives the TOA reflectance per spectral band.

For storage and visualization, the linear reflectance values can optionally be converted into a gamma-compressed digital number representation. This improves brightness and contrast for visualization by compensating for the nonlinear response of display systems [141]. Let $\rho \in [0, 1]$ denote a linear reflectance value. A gamma-encoded value ρ_γ is computed with Equation 5.73,

$$\rho_\gamma = \rho^{1/\gamma}, \tag{5.73}$$

with $\gamma = 2.2$ in the current implementation, matching the 8-bit image scale. The corresponding 8-bit pixel value PV is then obtained with Equation 5.74,

$$\text{PV}_{255} = \text{round}\left(255\, \rho_\gamma\right), \tag{5.74}$$

and clipped to the interval $[0, 255]$. When loading RGB PNG images as input, the inverse transformation can be applied to recover the linear reflectance values.

Because 8-bit COCO images were used as inputs for the pre-trained deep learning models, the gamma-corrected reflectance images are also used for the final model training to maintain dimensional consistency with the pre-trained layers. In operational applications, however, raw DN, radiance, or non-gamma-corrected reflectance data may be preferred to minimize onboard processing delays. This would likely require additional fine-tuning on a broader range of DN, radiance, and reflectance inputs to adapt the model to those feature domains.

The aim of this work is to provide an initial estimate of AI-Based Tip and Cue performance, instead of fully optimizing the preprocessing and model adaptation pipeline. This optimization therefore falls outside the scope of the present study, and the uncertainty introduced by not performing it should first be isolated. For this reason, gamma-scaled values are used for the first performance quantification.

Although gamma-corrected images may be more favorable for detection algorithms because they emphasize contrast, models operating on radiance or reflectance data could also be improved through more extensive optimization and fine-tuning prior to deployment. The resulting performance should therefore be interpreted as an initial estimate of AI-Based Tip and Cue off-nadir detection performance, rather than as a fully optimized upper bound.

5.3.4. Rendering and Post Processing

Integrator and Sampler Selection

To complete the render in Mitsuba, an integrator and sampler must be selected. The integrator determines the approach for solving the light transport equation, specifying which light transport paths are simulated and how radiance contributions are accumulated. The sampler determines where the samples, for which the light path is computed, are placed in the sampling domain.

Conceptually, the rendered pixel value can be written as a Monte Carlo estimate of a light transport integral. Let I be the radiometric pixel measurement (radiance or band-integrated radiance). The exact quantity can be expressed as an integral over the high-dimensional path space Ω,

$$I = \int_{\Omega} f(\mathbf{x})\, d\mathbf{x}, \tag{5.75}$$

where variables $\mathbf{x}$ represents a light path (including pixel position, lens sample, scattering directions, and wavelength samples), and $f(\mathbf{x})$ specifies its radiometric contribution.

Since Equation 5.75 cannot be evaluated analytically, Mitsuba approximates it using Monte Carlo integration. The estimator for the pixel value is therefore computed as an average over N sampled light paths as given in Equation 5.76:

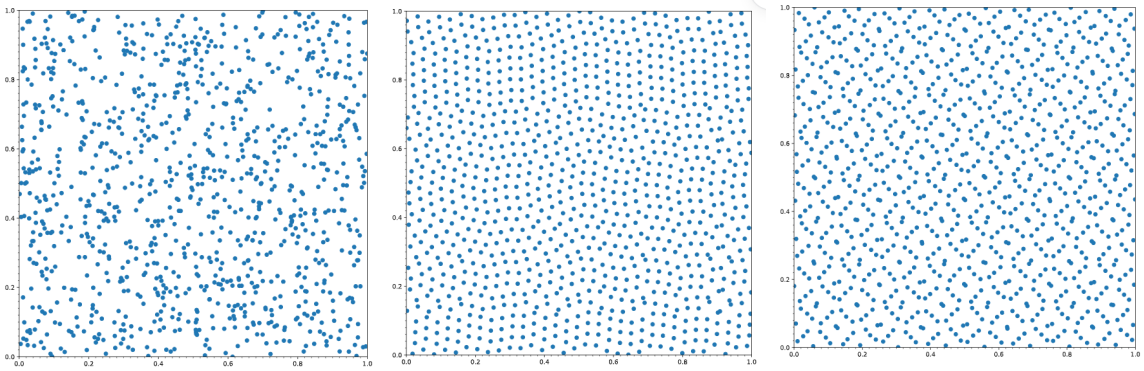
$$\hat{I} = \frac{1}{N} \sum_{k=1}^{N} \frac{f(\mathbf{x}_k)}{p(\mathbf{x}_k)}, \tag{5.76}$$

where $\mathbf{x}_k$ are samples drawn from a probability density function $p(\mathbf{x})$ defined by the integrator and sampler. The samples-per-pixel (SPP) parameter corresponds to N in Equation 5.76. Increasing N reduces estimator variance and therefore image noise, at the cost of increased computational time.

Integrator choice Mitsuba provides multiple integrators [129]. The `direct` integrator estimates only direct illumination, with a single bounce, and is efficient for scenes where indirect light transport, from for example scattering, is negligible. The `path` integrator is a general-purpose path tracer that supports direct and indirect illumination. It does so by simulating random walks starting from the sensor, making it a robust option when multiple scattering interactions contribute. Volumetric variants (`volpath`, `volpathmis`) extend this to participating media, which is not required here because the atmosphere is represented through external SPDs instead of volumetric rendering. Debug integrators like `depth` and `aov` return buffers and are mainly useful for verification rather than dataset generation.

Given the objective of radiometrically consistent rendering under both direct solar and diffuse sky illumination, as described in subsection 5.3.3, including multiple scattering between the ocean surface and the environment, the `path` integrator was selected for the AI-based Tip and Cue rendering pipeline [129].

Sampler choice The sampler determines how samples are distributed over the unit hypercube before being mapped to pixel positions, wavelengths, and scattering events [129]. Figure 5.28 visualizes different sampling strategies.



(a) Independent sampler **(b)** Multi-jittered sampler **(c)** Low-discrepancy sampler

Figure 5.28: Example 2D projections of sample distributions in the unit square for different sampling strategies. The independent sampler shows clustering, while low-discrepancy and multi-jittered samplers produce more uniform coverage, reducing variance in the Monte Carlo estimator. [129]

The `independent` sampler produces pseudorandom samples and is simple and unbiased, but typically converges slower due to sample clumping. The `stratified` sampler reduces variance by dividing the domain into a discrete number of strata and produces a sample within each one of them, but its 1D projections can still exhibit clustering. The `multijitter` sampler improves this in both 2D and 1D projections by first placing samples in a canonical arrangement and then shuffling the x-coordinate of the samples in all columns and the y-coordinate in all rows, which reduces variance for anisotropic integrands. The `orthogonal` sampler generalizes this idea to higher dimensions and aims to maintain good stratification for multiple projections. The `ldsampler` uses a low-discrepancy sequence with scrambling and is designed as a fast general-purpose option that often converges faster than purely pseudorandom sampling at the same SPP.

At infinite sample counts, all samplers converge to the same solution of the light transport integral. With finite sample count, it holds that for a higher sample count, the variance decreases, and the pixel value fluctuates less around the true solution. At the same time, increasing the sample count N leads to a linear increase in runtime. Given the goal to optimize for accuracy (low variance in the band-integrated radiance from Equation 5.38, and consequently in the reflectance from Equation 5.62) and to minimize computational cost, the `multi-jittered` was selected.

It should be noted that this selection could be argued in more detail and supported by a dedicated quantitative comparison of sampler-induced variance in the rendered radiance. However, in the present application the remaining Monte Carlo variance is expected to be small relative to the absolute magnitude of the band-integrated radiance, which will be verified by increasing sampling count in section 6.2. For this reason, an in-depth sampler study is considered outside the scope of this work.

Outer Area Masking

After rendering the scene, the sky is applied as a constant emitter as defined in subsection 5.3.3. Where the DEM mesh ends, the environment emitter still contributes radiance, producing a surrounding background region that does not correspond to the textured ocean surface as shown in Figure 5.29. This region is not physically meaningful for the dataset and may confuse the Deep Learning models. Therefore, a footprint mask is constructed and applied so that all pixels outside the DEM projection are set to zero for each rendered band.

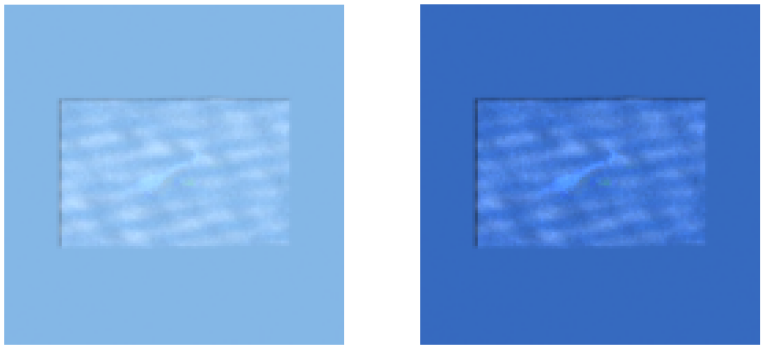


Figure 5.29: Rendered scene before masking. Left: band-integrated radiance image after gamma correction from Equation 5.38. Right: corresponding TOA reflectance after gamma correction from Equation 5.62. The surrounding blue region originates from the constant sky emitter and does not correspond to the DEM footprint. *Satellite image © 2022 Maxar Technologies.*

The footprint mask is derived from a black-reference render. This render is performed using the same geometry and camera configuration, but with a fully black texture as the diffuse input. In this reference image, pixels belonging to the projected DEM remain black given no reflection, while pixels outside the projected mesh are dominated by the uniform background. By thresholding the black-reference render, a raw background mask is obtained that separates the projected DEM footprint from the outer region.

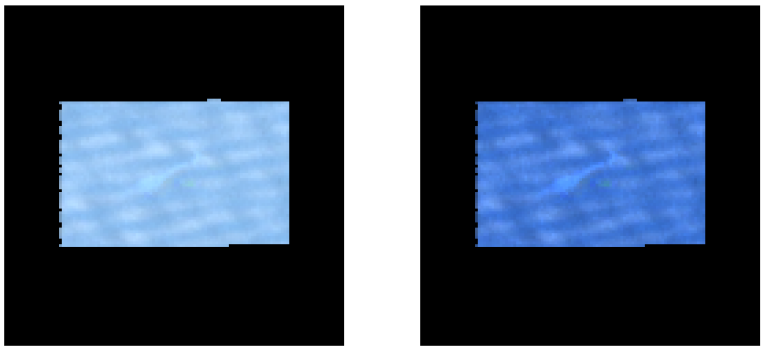


Figure 5.30: Row-wise post-processing of the raw background mask, obtained by thresholding the black-reference render. It shows the irregular edge artifacts and isolated pixels. Left: band-integrated radiance, right: TOA reflectance, both after gamma correction. *Satellite image © 2022 Maxar Technologies.*

To make the footprint stable against small holes and irregular edges caused by sampling noise and resampling artifacts, as shown in Figure 5.30, the raw mask is post-processed row-wise. First, the first and last rows containing background pixels are used to restrict the vertical extent of the candidate region. Within this vertical interval, rows are kept only if a sufficiently large fraction of pixels in that row are classified as background, which removes isolated background pixels that occur inside the footprint. For the remaining rows, the left and right background extents are estimated from the first and last background pixel in the row, and a rectangle is then fitted around the image region. This produces a binary mask that contains only full rows and a single contiguous footprint region, avoiding partial-row masking that would introduce noisy edges.

The resulting binary footprint mask is stored and applied to all rendered outputs. For each band, both the band-integrated radiance image from Equation 5.38 and the corresponding TOA reflectance image from Equation 5.62 are multiplied by this mask so that pixels outside the DEM projection are set to zero. The effect of the black-reference render and the extracted footprint are illustrated in Figure 5.31.

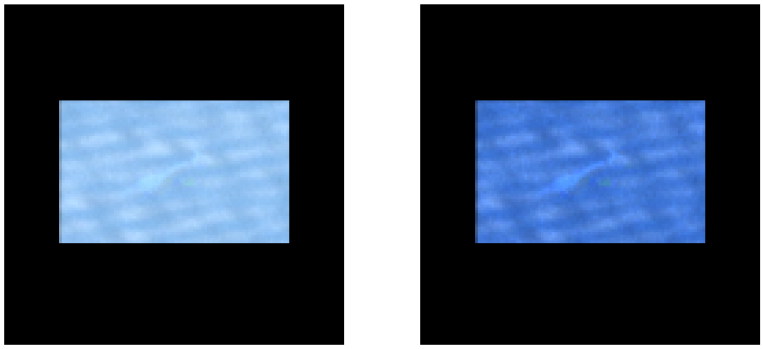


Figure 5.31: Black-reference render used for the resulting masked output after edge post-processing. Pixels outside the DEM projection are set to zero for both radiance (left) and reflectance (right), ensuring that only physically meaningful ocean surface regions are retained. *Satellite image © 2022 Maxar Technologies.*

Rendering Outputs

Figure 5.32, Figure 5.33, and Figure 5.34 show rendering outputs for three different scenes. Each row illustrates the full processing chain: the original RGB PNG image, the geometrically reprojected texture under constant illumination, the physically rendered band-integrated radiance including sun glint, the reflectance without glint, which is purely diffuse, and the combined reflectance including glint. All reflectance images are gamma corrected according to Equation 5.73 for visualization.

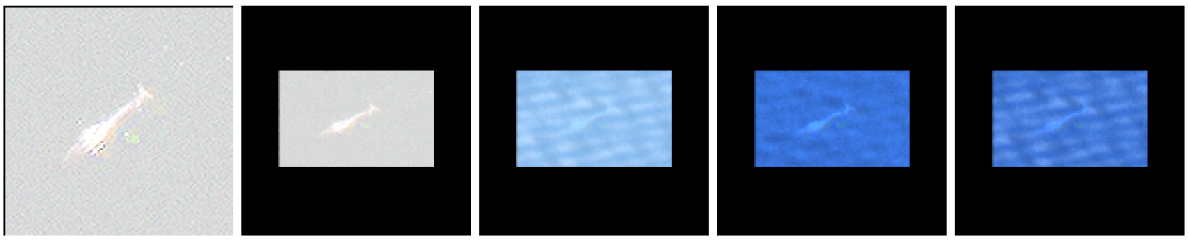


Figure 5.32: Rendering result for scene 1. From left to right: original RGB PNG image, reprojected texture under constant illumination, rendered band-integrated radiance including sun glint, reflectance without glint (diffuse only), and combined reflectance including glint. Radiance and reflectance images are gamma corrected for visualization. *Satellite image © 2022 Maxar Technologies.*

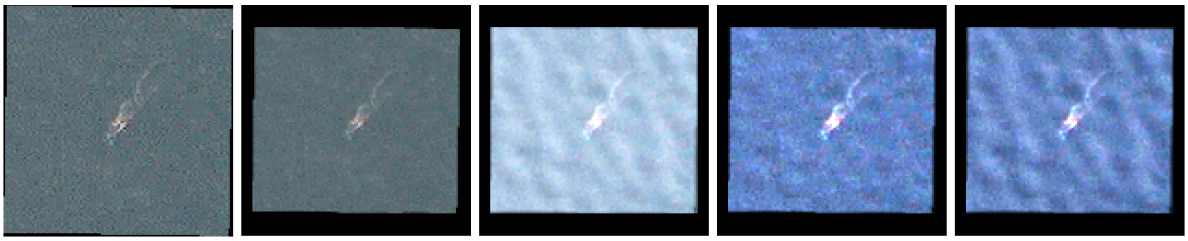


Figure 5.33: Rendering result for scene 2. From left to right: original RGB PNG image, reprojected texture under constant illumination, rendered band-integrated radiance including sun glint, reflectance without glint, and combined reflectance including glint. *Satellite image © 2022 Maxar Technologies.*

Figure 5.34: Rendering result for scene 3. From left to right: original RGB PNG image, reprojected texture under constant illumination, rendered band-integrated radiance including sun glint, reflectance without glint, and combined reflectance including glint. *Satellite image © 2022 Maxar Technologies.*

It can be observed that the reprojected constant-light image preserves the geometric transformation introduced by the off-nadir viewing configuration while maintaining uniform radiometric conditions.

The radiance images show brightening due to the microfacet-based sun-glint term defined in subsection 5.3.3, with an intensity magnitude specific to the relative solar and viewing geometry.

When converting to reflectance using Equation 5.62, the overall magnitude becomes normalized with respect to solar irradiance and solar zenith angle, resulting in a darker but geometrically consistent representation. The reflectance without glint isolates the diffuse component from the texture-based Lambertian term, while the combined reflectance shows the additive glossy contribution from the rough dielectric microfacet model.

From the three examples, it is visible that the strength and spatial distribution of the glint pattern vary depending on the observation geometry, where when the specular reflection direction approaches the sensor viewing direction, strong localized brightening occurs. When the relative configuration deviates from the specular alignment, the glint contribution diminishes and the reflectance becomes dominated by the diffuse texture component.

5.3.5. Patch Creation

The whale images from the Whales from Space dataset [100] are already provided in patch format, with dimensions varying between 128 x 128 and 145 x 145 pixels. Annotations are provided in the form of COCO bounding box and segmentation masks [142]. While the initial annotations were not always fully reliable, including missing annotations or an object offset, a tool was built to visually verify and correct the annotations if needed, as shown in Figure 5.35. It allows to re-annotate the segmentation mask by clicking points surrounding the geometry, after which a bounding box is automatically drawn edging the segmentation mask. With 633 images in total and a display time of 1 second each, it was possible to verify the correct implementation of the annotations for each image individually.

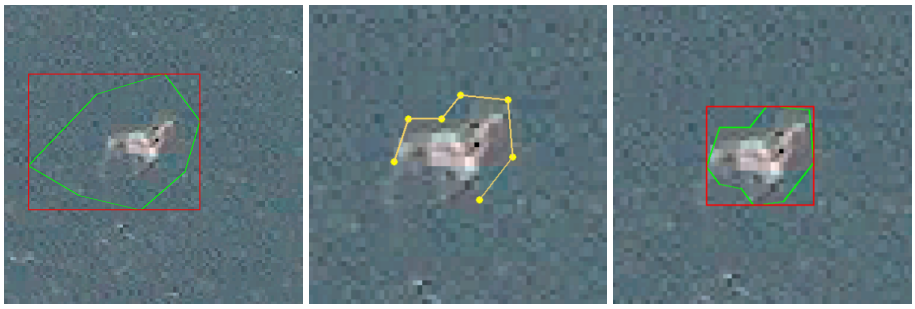


Figure 5.35: Visualization of the annotation correction tool. The left image shows the original annotation. If it is desired to correct it, the segmentation correction tool in the middle images allows to draw a new segmentation mask. Upon right click, a bounding box is drawn and the annotation is corrected as shown in the right figure. *Satellite image © 2022 Maxar Technologies.*

While in principle these patches can be loaded into the rendering pipeline to generate the full dataset, this would not reproduce reliable detection outcomes. This is because most patches contain a whale that is

centered in the middle, making it straightforward for the model to learn the whale location, leading to an overestimated model performance.

Therefore, the input images have to be cropped, both to randomize the location of the whale with respect to the image bounds, and to generate negative class patches as well containing ocean surface only.

To do so, first, a fixed window size for the new patches has to be defined. Given the smallest dimension is 128 x 128, which has to be cropped to both ocean and water with a whale in the middle, and aimed powers of 2 to better comply with standard model input dimensions, a window size of 64 x 64 was selected for the patch generation.

Then, by specifying a pixel location (p_x, p_y) within the original image, it can be considered as the upper left corner of the patch window. By randomly drawing a value for p_x and p_x within the upper left quadrant of the image, bigger than the dimension - 64, the placement location of the patch is randomized.

After that, it should be known whether the patch region selected contains a whale or not. Therefore, the COCO annotations were loaded to examine whether the patch contains a full whale, ocean, or a fraction of the whale, for which the fraction with respect to the original mask was also computed. With that, it became possible to generate only full whale patches, half-whale patches, or ocean patches upon user request.

After passing the request for full, half, or ocean, margins were added to increase the likelihood of a successful ocean patch generation. Because the whale is always in the middle of the image, it would barely make it impossible to successfully generate ocean patches without having a few whale pixels in the corner. Therefore, it was specified that a full whale counts as 99% of the whale segmentation included in the image, and ocean is reasonable if less than 10% of whale is included in the image.

With a typical ground sampling distance of 0.31, and whales of around 10 m long and 1m wide, it would mean that a whale object typically occupies 3 x 30 pixels = 90 pixels. 10% would mean that 9 whale pixels are allowed on the borders, on a full image of 64 x 64 = 4096 pixels, which is considered reasonable. Also, because the image borders are often partially cropped in the process of masking the diffuse sky radiation as described in subsection 5.3.4, this allowance of a few whale pixels on ocean patches is not expected to confuse or notably affect the detection model, which was later evaluated in the results.

Given those allowance boundaries, after a given request to either generate a full whale, half whale, or ocean, the patch region is sampled. To do so, an algorithm randomizes original image coordinates p_x and p_y many times, then verifies whether the patch contains the whale fraction as requested, and continues sampling at different locations until a successful match is found or the maximum number is reached. While this can be optimized, it is a computationally cheap operation, so no further effort was put into improving this operation.

Example patches are shown in Figure 5.36

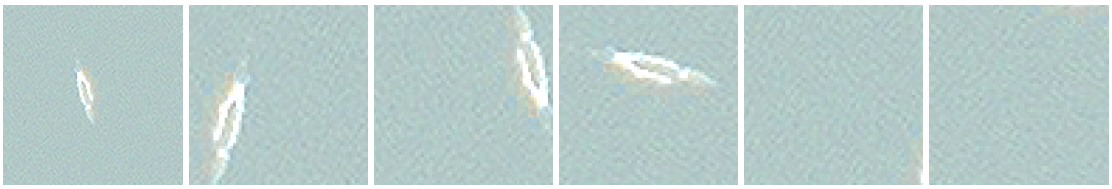


Figure 5.36: Result of the cropped patches. A window of 64 x 64 is cropped out of the original image, containing either a full whale, fraction of a whale, or ocean depending on user request. This patch is rotated and mirrored to ensure variety within the dataset. *Satellite image © 2022 Maxar Technologies.*

Off-Nadir Annotation Translation

After a patch is successfully generated, it is loaded into the rendering pipeline as described in section 5.3 and the off-nadir image is generated for the given satellite and target positions. Next, the patch annotations need to be translated. First to cropped patched coordinates, and after that to off-nadir angles.

To translate the annotations from the original image to the cropped patch coordinates, a linear translation is applied to the bounding box and segmentation mask based on the patch position.

For rotation, the bounding box and segmentation points are rotated by multiples of 90 degrees, for which, with local pixel coordinates x and y, the pixel remapping equations are given in Equation 5.77.

$$(x', y') = \begin{cases} (-x, -y) & \text{if } 180° \text{ rotation} \\ (y, -x) & \text{if } 90° \text{ clockwise} \\ (-y, x) & \text{if } 90° \text{ counterclockwise} \end{cases} \tag{5.77}$$

For mirroring, depending on the axis, the remappings are in Equation 5.78:

$$(x', y') = \begin{cases} (w - x, y) & \text{if horizontal flip} \\ (x, h - y) & \text{if vertical flip} \end{cases} \tag{5.78}$$

The result was visually verified in Figure 5.37, showing how the annotations are successfully translated.

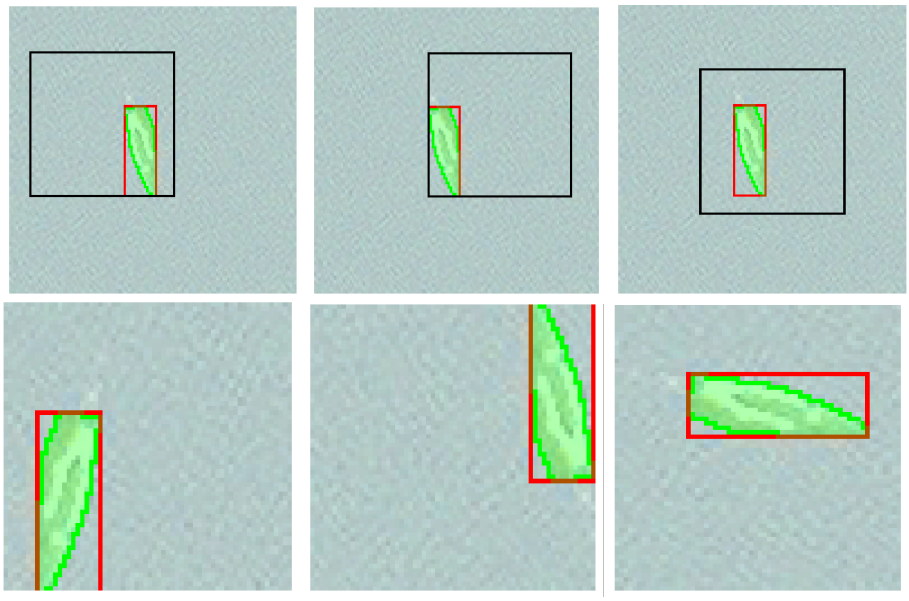


Figure 5.37: Visualization of the translated annotation masks. The top row represents the original image and the crop patch region of 64 x 64 pixels. The bottom row shows the new patch, after cropped, rotated, and optionally mirrored operation on the original patch. *Satellite image © 2022 Maxar Technologies.*

When loading the patch into the offnadir image rendering pipeline, its annotations are not automatically translated to the render as well. To do so, there would be two options. First, based on the scene and sensor geometry, geometrically translate the nadir annotation coordinates to the scene coordinates. However, the translation matrices are not provided by Mitsuba, meaning that it would be an effortful operation to derive and verify these by hand, falling outside the scope of this work.

The other option is, because these translations are solved internally within Mitsuba, to use the rendering pipeline to derive the new annotation position. Thus, a new image is generated where each bounding box point is displayed with a unique color ID. After that, the image is loaded into the Mitsuba rendering pipeline, but this time, not outputting radiance and reflectance, but operating in RGB mode. In this way, the original colors from the annotation points are preserved in the off-nadir render. With the annotation off-nadir render at the same satellite and target location as the off-nadir image, the pixel locations of the colored points can be derived, and based on their remapping, the bounding box in the offnadir render can be extracted. The approach is illustrated in Figure 5.38.

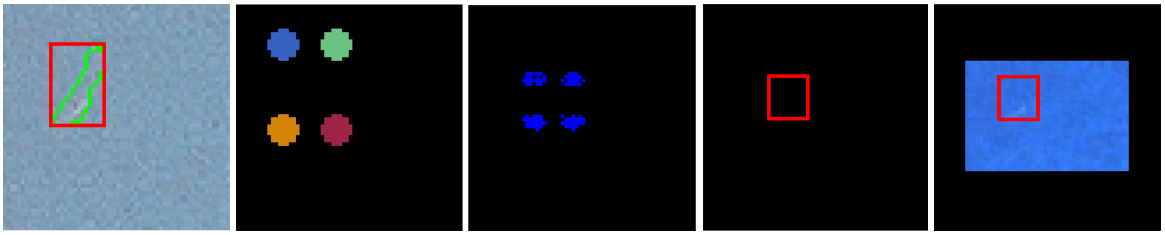


Figure 5.38: Visualization of the bounding box translation to off nadir angles. From left to right: original patch and annotations, bounding box corner point mapping, generated render of bounding box corner point mapping at off nadir angle, bounding box extraction, final result. *Satellite image © 2022 Maxar Technologies.*

The same approach was tried with the segmentation mask. However, the segmentation mask consists of way more than four points, sometimes with close proximity. Therefore, the points should either be remapped individually, making it a very computationally heavy operation which was aimed to be avoided. Therefore, an alternative approach was chosen where the segmentation mask was filled and colored white, while non-segmentation mask is set to black. That image is then loaded into the off-nadir rendering pipeline. After the render is being made, the pixels surrounding the white region are extracted and defined as the new segmentation mask. The result is shown in Figure 5.39.

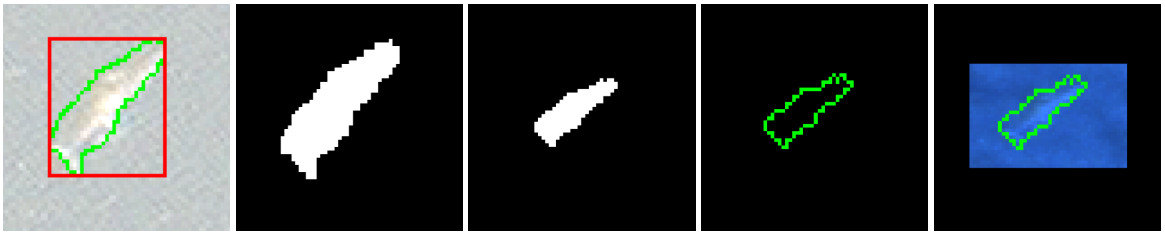


Figure 5.39: Visualization of the segmentation mask translation at off-nadir angles. From left to right: original patch and annotations, filled original segmentation mask, off nadir render of filled segmentation mask, mask extraction, final result. *Satellite image © 2022 Maxar Technologies.*

After the annotation generation, both bounding box and segmentation mask were exported to COCO format and stored in the .json annotation file.

Typically, when augmenting a dataset with rotating and mirroring images only, random noise should be added to prevent the model from learning the pattern of the object by heart. In this case, because the image is translated to off nadir angles and a full new segmentation mask is derived, it already displays a different pattern. Therefore, this pipeline is highly beneficial for turning a small dataset into a bigger one, when limited training data is available.

5.4. Onboard Detection

Following the patch generation in section 5.3, this section describes the onboard detection component used to evaluate target recognition performance within the AI-based Tip and Cue framework. First, the dataset generation process is introduced, including the construction of off-nadir image samples based on the simulated observation geometry in subsection 5.4.1. Next, the DEIMv2 detection model is presented in subsection 5.4.2, and hyperparameters are optimized in subsection 5.4.3. The model will provide the detection-level input required for the mission-level analysis in section 5.5.

5.4.1. Dataset Generation

Input Dataset

A representative input dataset is required to evaluate onboard detection performance within the AI-based Tip and Cue framework. For that purpose, the Whales from Space dataset [100], as introduced in Chapter 3, was selected as the baseline input dataset.

The Whales from Space dataset contains 633 annotated whale instances identified from 6,300 km^2 VHR

satellite imagery. The imagery was acquired by WorldView-3 (WV3), WorldView-2 (WV2), GeoEye-1 (GE1), and QuickBird-2 (QB2) across multiple study regions worldwide, including Argentina, New Zealand, South Africa, the United States, and Mexico. The dataset includes four whale species: southern right whale (*Eubalaena australis*), humpback whale (*Megaptera novaeangliae*), fin whale (*Balaenoptera physalus*), and gray whale (*Eschrichtius robustus*) [100].

Sample patches from different locations are shown in Figure 5.40. A more extensive overview of the dataset is provided in Appendix D. The study locations, source satellites, average ground sampling distance, species, and image counts are summarized in Table 5.6.

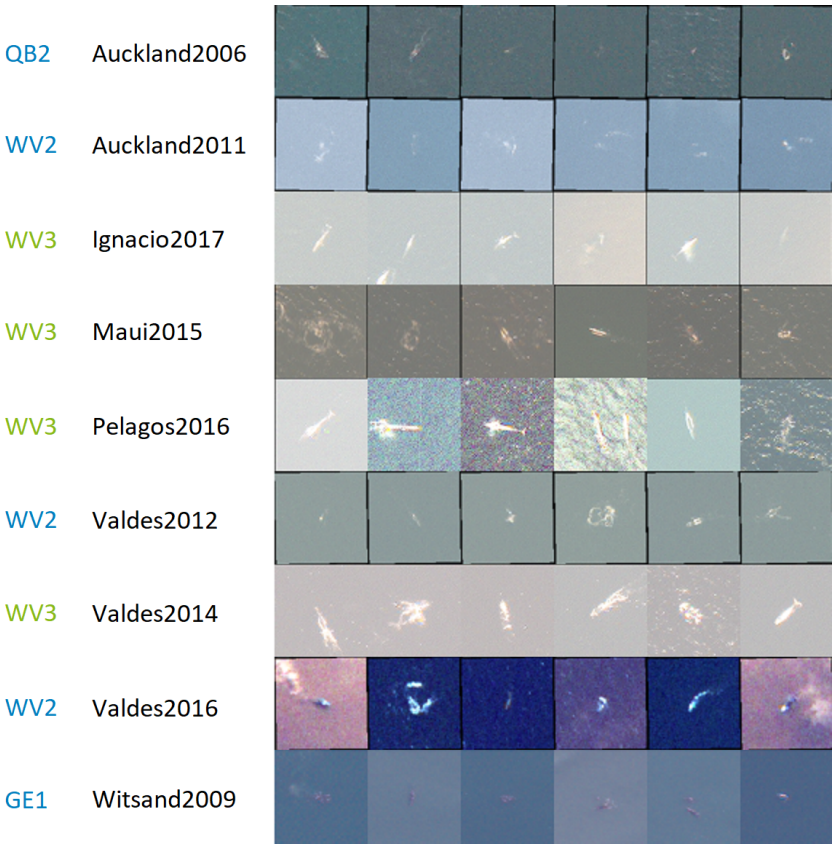


Figure 5.40: Example patches from the Whales from Space [100] dataset. *Satellite image © 2022 Maxar Technologies.*

Table 5.6: Overview of the study locations, satellite sources, average ground sampling distance, whale species, common species names, approximate body lengths, and image counts.

Location	Satellite	Avg. GSD [m]	Scientific Name	Common Name	Length [m]	Image Count
Auckland2006	QB2	0.54	*Eubalaena australis*	Southern right whale	14–15	69
Auckland2011	WV2	0.46	*Eubalaena australis*	Southern right whale	14–15	34
Ignacio2017	WV3	0.30	*Eschrichtius robustus*	Gray whale	13–14	80
Maui2015	WV3	0.30	*Megaptera novaeangliae*	Humpback whale	14–15	56
Pelagos2016	WV3	0.30	*Balaenoptera physalus*	Fin whale	20–22	34
Valdes2012	WV2	0.45	*Eubalaena australis*	Southern right whale	14–15	84
Valdes2014	WV3	0.30	*Eubalaena australis*	Southern right whale	14–15	59
Valdes2016	WV2	0.50	*Eubalaena australis*	Southern right whale	14–15	129
Witsand2009	GE1	0.50	*Eubalaena australis*	Southern right whale	14–15	88

Given that the samples originate from different satellite systems, both the source-image ground sampling distance and the original sensor characteristics vary across locations. This is relevant for the present work because the source image is used as the underlying texture in the off-nadir rendering pipeline.

Consequently, the spatial detail preserved in the rendered output remains limited by the spatial detail available in the original patch. Redefining the Mitsuba sensor therefore does not recover object information that is absent in the source imagery.

This effect is also visible in Figure 5.40: the WorldView-3 samples at 0.30 m ground sampling distance contain visibly more whale detail than the WorldView-2, GeoEye-1, and QuickBird-2 samples at 0.45-0.54 m. This matters because the off-nadir limit is ultimately in charge of how much object structure remains available after the additional geometric degradation introduced by off-nadir viewing.

At the same time, the dataset does not only vary in source resolution, but also in whale size. Most locations contain whales of approximately 13-15 m, whereas Pelagos2016 contains fin whales of approximately 20-22 m. These larger whales are expected to remain detectable under stronger degradation because they occupy more pixels in the original image. Therefore, the resulting off-nadir limit cannot only be interpreted as a function of ground sampling distance alone. It is a combined effect of source-image resolution and whale size, in addition to residual source-image characteristics.

To account for the dominant spatial-resolution difference while retaining sufficient samples per group, the off-nadir analysis is performed for two source-resolution bins. The first bin contains the WorldView-3 imagery at 0.30 m ground sampling distance. The second bin groups WorldView-2, GeoEye-1, and QuickBird-2 imagery, ranging from 0.45 to 0.54 m. This grouping is considered appropriate because, upon visual inspection, whales in the latter three sources appear approximately similar in size, whereas the WorldView-3 imagery consistently contains more spatial detail. The second bin should therefore be interpreted as a practical lower-resolution source group, not as a single-GSD experiment.

Given different satellites, the images are also captured with different spectral characteristics. In the off-nadir rendering pipeline as described in section 5.3, the input image serves as the baseline texture that is applied to the DEM file. Because the original satellite imagery is provided in RGB digital numbers, it does not directly represent physical reflectance. Therefore, the image is first rescaled to a reflectance representation, after which a new Mitsuba sensor is specified and the image is rendered according to that sensor specification. In that sense, the source image mainly provides the spatial texture basis, while the final radiometric response is determined by the rendering pipeline and the sensor model defined for the experiment.

Synthetic Tip Data Exploration

While Tip satellite data is limitedly available, synthetic dataset generation methods, based on the Whales from Space Dataset, have been explored.

An initial objective was to model detection for both Tip and Cue satellites, to simulate an end-to-end chain, where both the Tip and Cue satellite data could be loaded into the rendering pipeline as described in section 5.3 to achieve consistent radiometric conditions. Ideally, this would require labelled whale data at medium spatial resolution, from wide-swath sensors like Sentinel-2. However, no publicly available labelled whale dataset exists at this scale, and generating a representative dataset from manual annotations would exceed the scope of this work.

Downsampling very-high-resolution (VHR) Cue imagery to emulate Tip data was also considered. Due to the commercial nature of most VHR whale datasets, wide-area source imagery was not accessible.

Synthetic data generation was therefore evaluated to compensate for the lack of suitable wide-swath labelled imagery. Two approaches were considered: physics-based scene synthesis using SeaDroneSim [101], and Deep Image Blending (DIB) [143] for compositing actual whale instances into larger background images.

SeaDroneSim [101] provides a controllable simulation environment for maritime scenes, enabling the generation of synthetic whales with associated segmentation masks and adjustable environmental parameters. An example output is shown in Figure 5.41. While this offers flexibility in pose, scale, and scene composition, several limitations arise. First, synthetic scenes cannot reliably reproduce real atmospheric variability, complex sea states, or partially submerged whale appearances. Second, radiometric characteristics remain model-dependent and are not tied to a specific satellite sensor response. Third, the simulator is trained and validated primarily on VHR imagery, making extrapolation to realistic wide field-of-view Tip satellite conditions uncertain. As the objective of this work is to quantify physically consistent off-nadir

degradation rather than to introduce additional modelling assumptions, SeaDroneSim was not adopted for dataset generation.

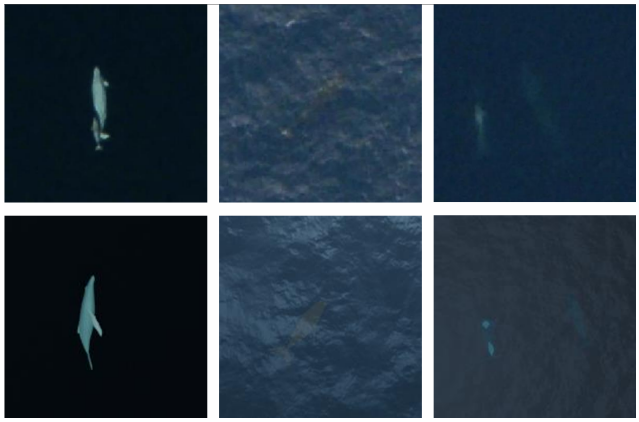


Figure 5.41: Example patches generated by SeaDroneSim2. The first row presents real satellite images while the second row contains the artificially generated images from SeaDroneSim2 [101].

Deep Image Blending (DIB) was subsequently tested to insert labelled whale patches into larger ocean background imagery. Given an input background image, whale crop, segmentation mask, and insertion coordinates, as shown in Figure 5.43, DIB can generate a visually coherent composite. In this setup, a whale-containing patch was cropped from the original Sentinel scene and then pasted back into the same spatial location as shown in Figure 5.42, replacing the original background region so that the scene now contains a whale at that position.

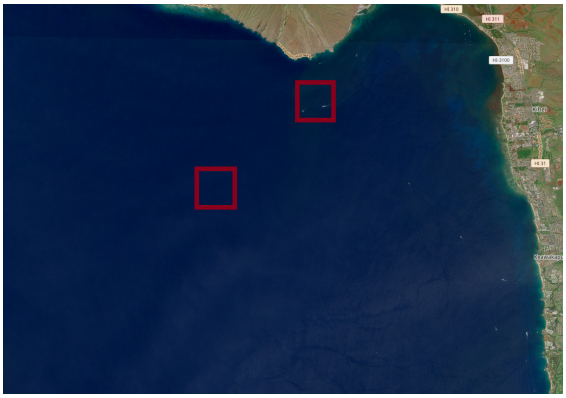


Figure 5.42: The full Sentinel-2 image with approximate location of the target patch extractions. [4]

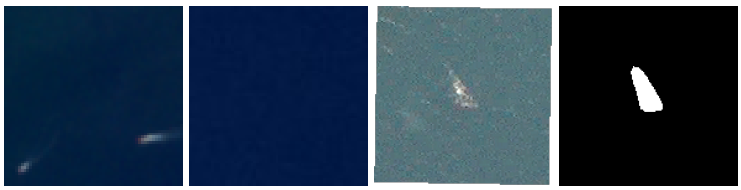


Figure 5.43: The inputs of the Deep Image Blending model, counted from left to right: either target patch with or without reference features (1 and 2), source image (3) [100], and its mask (4). *Satellite image © 2022 Maxar Technologies.*

[4] Copernicus Data Space Ecosystem, "Copernicus Browser," Available: `https://browser.dataspace.copernicus.eu/`. Accessed: Apr. 2, 2025.

Figure 5.44: The results of the Deep Image Blending model after the first pass for artificially generating Sentinel-2 images containing whales of different sizes. From left to right, whale size: 16, 8, 4, and 2 m. First row: patch that contains reference features, second row: patch that does not contain reference features.

Although the composites appear realistic at first inspection, quantitative evaluation revealed pixel intensity differences in the insertion region on the order of 30-80 digital counts on an 8-bit scale, as visualised in Figure 5.45. These discontinuities create artifacts that detection models potentially learn as shortcuts, reducing generalisation to real imagery. Furthermore, DIB operates exclusively in the visible spectrum. However, whale discrimination from vessels, rocks, or ice is expected to benefit from infrared contrast due to thermal and reflectance differences [144]. As no suitable infrared dataset was available within the project constraints, extending DIB to multispectral blending would not yield scientifically meaningful or practically extendable results for the intended Tip and Cue evaluation.

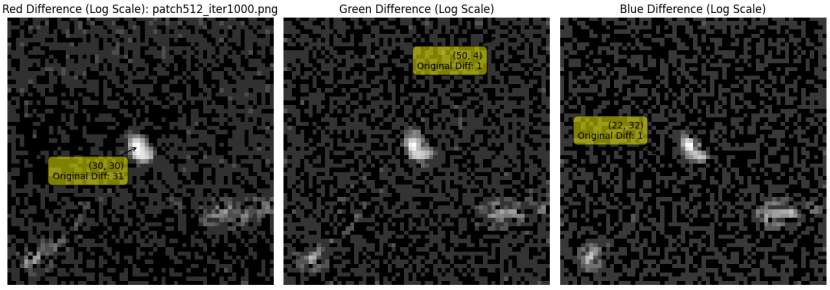


Figure 5.45: The differences between the artificially generated whale image compared to the original in red, green, and blue spectrum. Plots are made in logarithmic scale to better visualize small differences.

Given these geometric and radiometric inconsistencies, both synthetic approaches were considered unsuitable for establishing a physically controlled benchmark. The scope was therefore limited to the most critical component for AI-based Tip and Cue evaluation: quantifying the impact of off-nadir distortions on Cue satellite imagery.

Generation Method

To generate a dataset for Cue satellite imagery, first, patches have to be created. After loading the patch into the rendering pipeline and translating its annotations as described in section 5.3, the full dataset is generated.

The goal is to generate a dataset that contains an equal number of ocean and whale samples, with an even distribution of off-nadir angles, so that the off-nadir limit of the detection model can be fairly determined.

To sample images at a given off-nadir angle, also matching target and satellite position that is representative for the actual mission, orbital simulation results are available from section 5.2. These contain an overview of all detections within one simulated mission, for a given simulation configuration. Because within each simulation, the Cue satellite is tasked to capture at a specified off-nadir angle, ranging from

0 to 60 degrees with a step size of 5 degrees, there is plenty of data available from satellite and target position for a detection with specified off-nadir angle.

The detections from the simulation logging files, all representing one simulated mission each, were merged into a detection database specifying off-nadir angle, satellite position, target position, and time for a targetted mission. Then when specifying a nadir angle request, an instance matching the off-nadir angle in the simulation database is randomly picked, from which an actual satellite and target position and time were obtained.

Now, with 633 images from the Whales from Space dataset, the goal is to sample it at each off-nadir angle of 5 to 60 degrees, with a step size of 5, and extract both a whale and ocean image. From the 633 images, all having 12 off-nadir angles for both whale and ocean, it is expected to produce a dataset of 15,192 images in total, with 7,596 images per class.

It was also considered to model half-whales in the dataset as well. However, because the swath width of the sensor, based on WorldView-3, is 13.1 x 13.1 km, and a whale object is around 10 x 1 meter, it is deemed very unlikely that a whale would occur on the border of the satellite image in practice. Given that, and aiming to prevent additional complexity in the evaluation, it was decided not to include half-whales in the dataset, and stick with either full whale or full ocean samples only.

Looping through the 633 images, other image parameters are also randomized to increase patch diversity. For each sampling instance, the wind speed is drawn from a normal distribution between 2 and 12 meters per second [145], the DEM seed is randomized, meaning that each image contains different waves, the location of the patch cropping from the original is randomized so that each patch is cropped from a different location with respect to the original, the seed to obtain the satellite and target position from the mission database is also randomized, and the image is rotated 0, 90, 180, or 270 degrees, and either mirrored or not. With that, a final dataset is generated.

However, in some cases, the image generation does fail. One of them is when the whale in the middle is so big that no ocean region containing less than 10% whale can be extracted. Another option is when the off-nadir angles become so high, the resolution becomes so high that the whale object vanishes in one pixel, and the bounding box points cannot be separately mapped and distinguished. In the latter case, this happens after 55 - 60 degrees. It could be fixed by rendering each bounding box point separately, however, because the whale is already vanished to such an extent that it is not expected to be detected by the model anymore, no effort was put into realizing this fix.

When the image generation fails, the automated pipeline skips over it. Therefore, it should be noted that even though the specification for an even off-nadir angle and class distribution, the actual dataset has to be inspected afterward for its failed instances.

Generation Result

Executing the automated pipeline and merging the whale and ocean class datasets and COCO annotations, the final dataset was generated. It contains 13,925 images, of which 7,061 whale and 6,864 ocean class. Samples from the dataset, both with and without annotations, is shown in Figure 5.46.

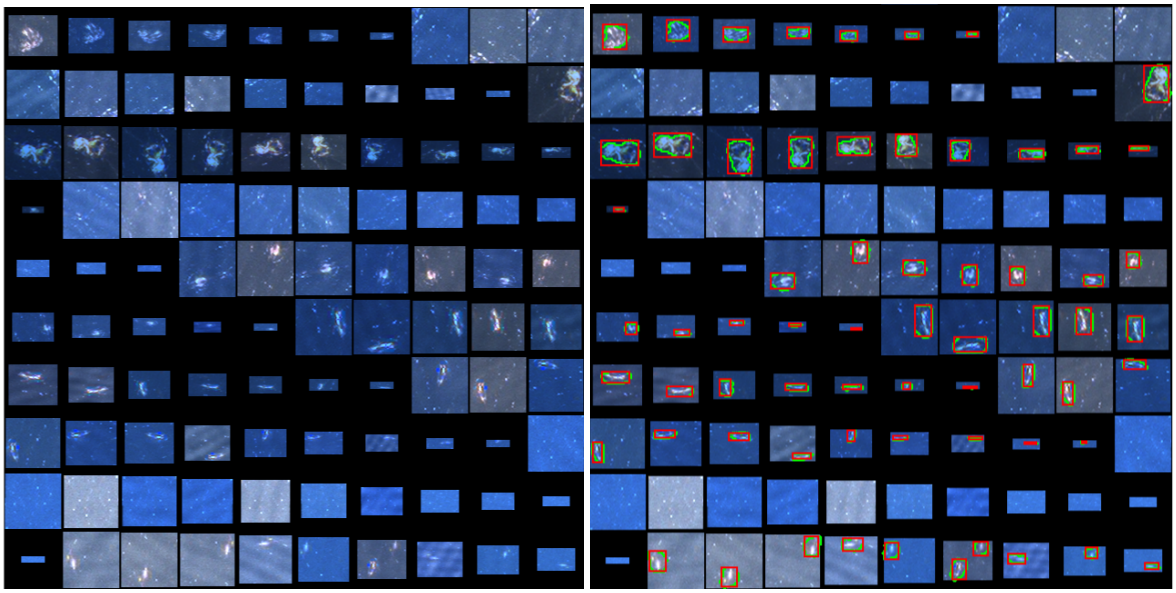


Figure 5.46: Example images of the generated dataset with geometric and radiometric effects using the rendering method as described in section 5.3. On the left, without annotations, on the right with annotations for the same images. *Satellite image © 2022 Maxar Technologies.*

To compare the results, models were also trained using datasets where radiometric modeling was not applied, as well as datasets where both geometric and radiometric effects were excluded, which is the standard practice in satellite deep learning. These additional datasets were generated with the same patch cropping and mirroring techniques to ensure comparability. Note that these cropped nadir patches also represent the inputs for the off-nadir imaging rendering engine. The original patches, both without and with geometric off-nadir effect modelling, are shown in Figure 5.47 and Figure 5.48 respectively. More sample boards, displayed at larger size and also compared for different original locations, are included in Appendix D.

Note that they all three represent the same input image, where the radiometric effects were modelled for different orbital positions within the same satellite orbit. It was validated that these are radiometrically representative for actual orbital configurations in section 6.2.

This process expanded the dataset from 633 input patches to a diverse collection of 13,925 images (22X). In addition to the radiometric diversity, the off-nadir annotation resampling ensures that the segmentation mask in each image is not merely a linear translation or rotation. As a result, each sample has a unique spatial and radiometric pattern. This approach significantly enhances small satellite datasets, providing a substantial contribution to fields with limited data availability, such as whale monitoring, which is further discussed in Chapter 8.

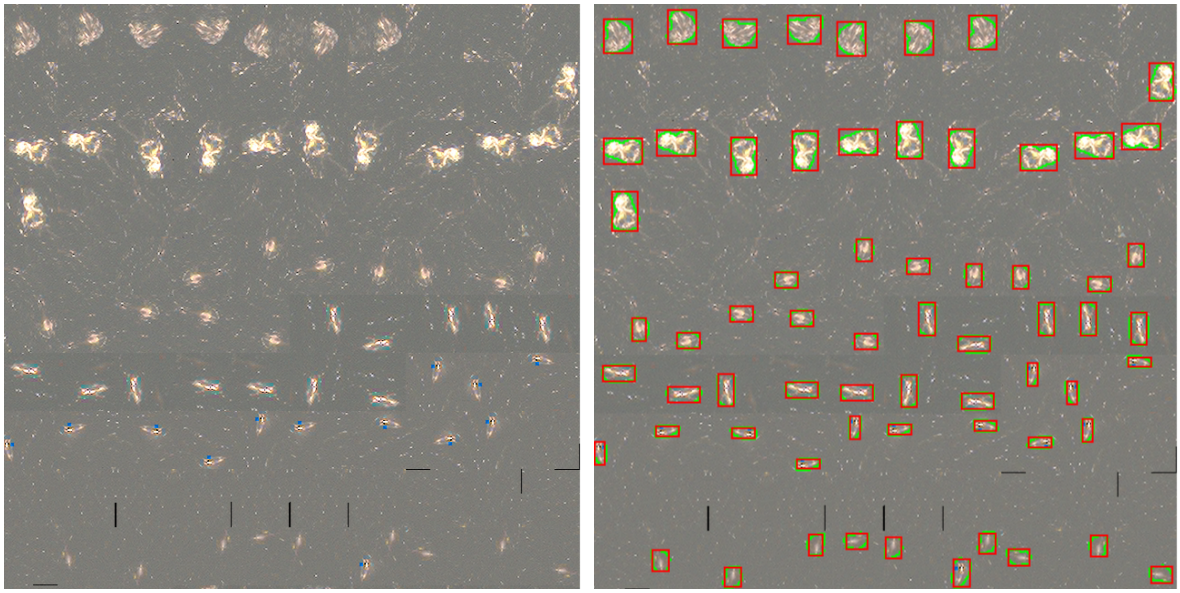


Figure 5.47: Example images of the generated dataset based on the original images. On the left, without annotations, on the right with annotations for the same images. *Satellite image © 2022 Maxar Technologies.*

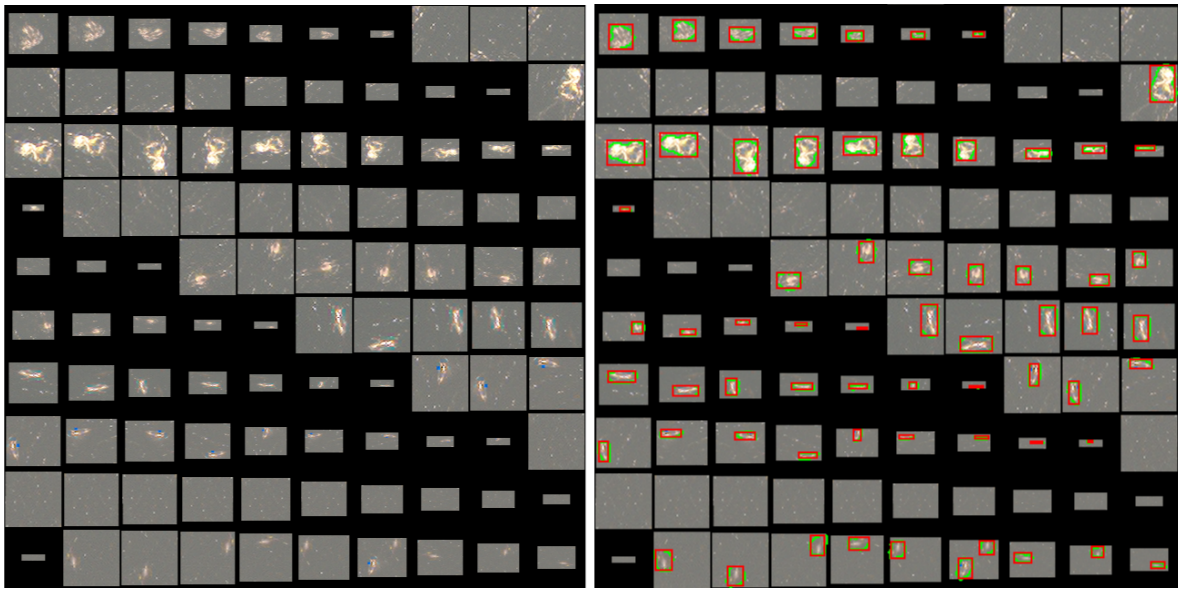


Figure 5.48: Example images of the generated dataset without radiometric effects. On the left, without annotations, on the right with annotations for the same images. *Satellite image © 2022 Maxar Technologies.*

The off-nadir distributions for each class are listed in Table 5.7. First, it is observed that the class distributions are not perfectly equal. From the estimated 7,596 images per class, as expected, the ocean shows the largest image decline. This is because sometimes it is not possible to crop out an ocean patch of 64 x 64 out of the image when the whale is too big. Still, the distribution of the classes deviates by less than 10%, with slightly more samples for the positive class, which is favoured as those are the features to learn.

It is also observed that the off-nadir distributions show a steep decline near 60 degrees for the whale class, from 550 to 389. As described in subsection 5.3.5, this is due to the bounding box translations, where they are rendered and remapped to their new off-nadir coordinates. However, at 60 degrees, the whale sample becomes so small that it occupies a single pixel in width, which makes the bounding box points overlap so that they cannot be resolved. This could be fixed by instead of remapping all four simultaneously,

remapping each point individually. However, given significant increase in computational cost, and with the goal to evaluate the maximum off-nadir angle, which is expected to be below 60 degrees, it was decided not to implement this fix. Still, the decline in positive samples / difference in class distributions has to be accounted for in the off-nadir angle evaluations.

Table 5.7: Counts of Whale, Ocean, and Combined Images per Off-Nadir Angle within the generated dataset.

Off-Nadir Angle (deg)	Whale Images	Ocean Images	Combined Images
5	615	572	1187
10	615	573	1188
15	613	570	1183
20	623	579	1202
25	609	565	1174
30	620	574	1194
35	610	570	1180
40	613	572	1185
45	609	572	1181
50	596	573	1169
55	549	572	1121
60	389	572	961
Total	**7061**	**6864**	**13925**

Table 5.8: Counts of Ocean, Whale, and Combined Images per Location within the generated dataset.

Location	Whale Images	Ocean Images	Total Images	Original Images
Auckland2006	790	828	1618	69
Auckland2011	361	408	769	34
Ignacio2017	923	863	1786	80
Maui2015	646	516	1162	56
Pelagos2016	374	300	674	34
Valdes2012	911	984	1895	84
Valdes2014	685	623	1308	59
Valdes2016	1396	1514	2910	129
Witsand2009	975	828	1803	88
Total	**7061**	**6864**	**13925**	**633**

Data Distribution

The Whales from Space dataset consists of data from nine distinct locations. To ensure a robust evaluation, the dataset is split into training, validation, and testing sets. Specifically, 5 locations are used for training, 2 for validation, and 2 for testing. The training and validation sets are randomly swapped across different locations to facilitate cross-validation and ensure that the model is independently trained on all locations. The final evaluation is conducted on data from two previously unseen locations, simulating real operational scenarios. This approach ensures that the model generalizes well and is not biased by location-specific characteristics.

5.4.2. Model Architecture

Next, a model must be selected, trained, and evaluated using the AI-Based Tip and Cue mission data.

Model Selection

For the Cue whale detection task, the DEIMv2 model was recommended by the European Space Agency ϕ-lab. Released in September 2025, DEIMv2 is a state-of-the-art model for real-time COCO detection. The S model is the first to surpass 50 AP on the COCO benchmark with fewer than 10M parameters, leading to latencies around 5 ms. It uses a DINOv3 backbone based on the Vision Transformer (ViT) family [146].

Comparison DEIMv2 vs YOLO26 To assess the performance of DEIMv2, the repository provides a comparison to other deep learning detection models. The comparison is illustrated in Figure 5.49 [146].

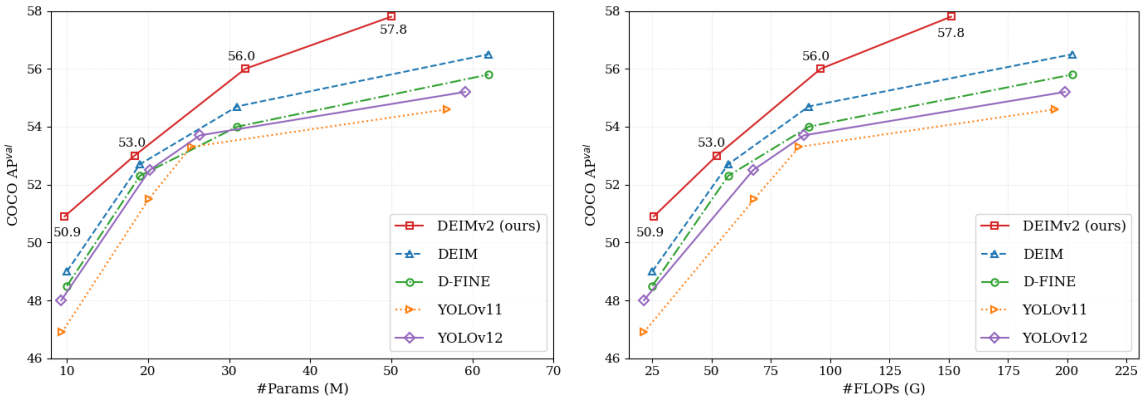


Figure 5.49: $AP_{50:95}$ performance comparison of DEIMv2 and other CNN- or transformer-based detectors for the number of parameters and GFLOPs. [146]

To keep up with the state-of-the-art, the latest YOLO model, YOLO26, was additionally compared to the DEIMv2 network. YOLO26 employs a single convolutional neural network to predict bounding boxes and class probabilities in one pass. The image is divided into an SxS grid, where each grid cell detects objects whose center falls within that grid, predicting a fixed number of bounding boxes, confidence scores, and class probabilities. Given its single pass and CNN architecture, YOLO is known for being fast and efficient, enabling real-time object detection [147].

Additional plots comparing the performance of YOLO26 and DEIMv2 are shown in Figure 5.50, based on the data from Table 5.9 [146], [147], [148].

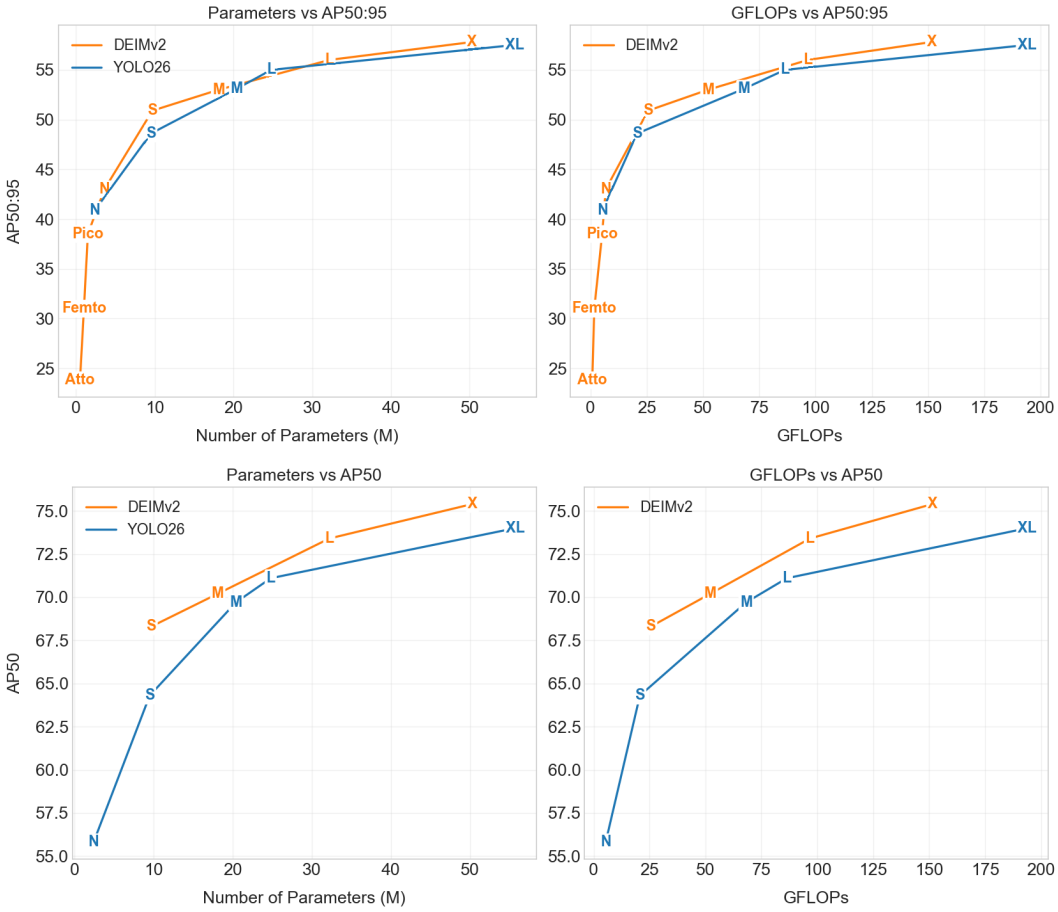


Figure 5.50: AP_{50} and $AP_{50:95}$ performance comparison of DEIMv2 and YOLO26 for the number of parameters and GFLOPs.

Table 5.9: YOLO26 and DEIMv2 Model Comparison. [146], [147], [148]

Model	Params (M)	GFLOPs	AP_{50}	$AP_{50:95}$	Size
YOLO26n	2.4	5.4	55.8	40.9	N
YOLO26s	9.5	20.7	64.3	47.8	S
YOLO26m	20.4	68.2	69.7	52.5	M
YOLO26l	24.8	86.4	71.1	54.4	L
YOLO26x	55.7	193.9	74.0	56.9	XL
DEIMv2-Atto	0.5	0.8	-	23.8	Atto
DEIMv2-Femto	1.0	1.7	-	31.0	Femto
DEIMv2-Pico	1.5	5.2	-	38.5	Pico
DEIMv2-N	3.6	6.8	-	43.0	N
DEIMv2-S	9.7	25.6	68.3	50.9	S
DEIMv2-M	18.1	52.2	70.2	53.0	M
DEIMv2-L	32.2	96.7	73.4	56.0	L
DEIMv2-X	50.3	151.6	75.4	57.8	X

At $AP_{50:95}$, the performance of DEIMv2 is similar to YOLO26. However, at AP_{50}, DEIMv2 outperforms YOLO26. Given that DEIMv2 is transformer-based and benefits from global reasoning, it generally requires more training data, which could be a concern for ensuring its performance. With a dataset of around 14,000 images, and prior work demonstrating successful ViT training for onboard evaluation in a similar scale [149], [150], it was determined that the generated dataset would be large enough for training a transformer-based model. Therefore, DEIMv2 was selected for the AI-Based Tip and Cue demonstration experiments.

DEIMv2 Architecture

DEIMv2 follows a design approach similar to RT-DETR, which integrates a backbone, a hybrid encoder, and a decoder in the object detection pipeline [83]. However, DEIMv2 introduces several optimizations to enhance its real-time performance. Its backbone design is shown in Figure 5.51.

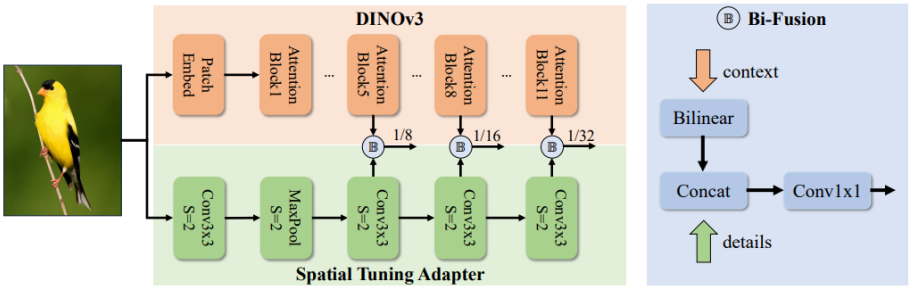


Figure 5.51: DEIMv2 architecture showing the integration of the DINOv3 backbone and the Spatial Tuning Adapter (STA) with the Bi-Fusion operator. [146]

Backbone For the larger variants (L, X), DEIMv2 employs DINOv3-pretrained Vision Transformers (ViT) as the backbone. DINOv3 is a pretrained version of ViT that benefits from a student-teacher learning process [151]. In this process, the teacher model is a large, pre-trained ViT model, while the student model is a smaller ViT model that learns to mimic the teacher's feature representations. This self-supervised learning process enables the student model to learn visual representations without relying on labeled data [152]. Specifically, ViT-Small and ViT-Small+ are used for the larger variants of DEIMv2, while the smaller variants S and M utilize ViT-Tiny and ViT-Tiny+ [146].

For the ultra-lightweight models, including Nano, Pico, Femto, and Atto, DEIMv2 utilizes HGNetv2 with depth and width pruning. This is designed to meet the strict resource budgets required for edge and mobile deployment while still achieving compatitive performance [153].

Spatial Tuning Adapter To handle the output from the DINOv3 backbone, DEIMv2 introduces the Spatial Tuning Adapter (STA), which is a convolutional neural network. STA converts the single-scale (1/16) output from DINOv3 into multi-scale features. This is achieved using bilinear interpolation across multiple ViT blocks (the 5th, 8th, and 11th), followed by the Bi-Fusion operator, which applies a 1x1 convolution to enhance the feature representation by adding fine-grained details.

Decoder The decoder in DEIMv2 is enhanced to refine object queries and predict bounding boxes and object classes. It incorporates SwiGLUFFN to improve the nonlinear representation capacity, in other words, to understand more complex patterns in data [154], and RMSNorm to stabilize and accelerate training [155].

Dense O2O Lastly, DEIMv2 introduces Dense O2O, an augmentation method building on previous DEIM work [156]. This method increases the number of object queries to improve supervision. By generating more training examples, it enhances the model's ability to learn finer details and improve detection accuracy. Additionally, DEIMv2 employs an object-level Copy-Blend augmentation, where new objects are added without backgrounds, and they are blended into the image. Unlike the traditional Copy-Paste method, Copy-Blend maintains the object edges, further enhancing the model's ability to learn accurate features. [146]

Loss Function The DEIMv2 model total loss function contains five components: Matchability-Aware Loss (MAL), Fine-Grained Localization Loss (FGL), Decoupled Distillation Focal Loss (DDF), Bounding Box Loss (L1), and Generalized Intersection over Union (GIoU) Loss, as defined in Equation 5.79.

$$L_{\text{total}} = \lambda_1 L_{\text{mal}} + \lambda_2 L_{\text{fgl}} + \lambda_3 L_{\text{ddf}} + \lambda_4 L_{\text{bbox}} + \lambda_5 L_{\text{giou}} \tag{5.79}$$

These loss components are weighted by the following values: $\lambda_1 = 1.0$, $\lambda_2 = 0.15$, $\lambda_3 = 1.5$, $\lambda_4 = 5.0$, and $\lambda_5 = 2.0$.

- Matchability-Aware Loss (MAL): MAL is a high-level loss that improves the matching process between predicted and ground truth bounding boxes. It adjusts the penalty for low-quality matches, which speeds up convergence and enhances detection accuracy [156].
- Fine-Grained Localization Loss (FGL): FGL is a finer loss that refines bounding box localization at a detailed level. It optimizes bounding box edge predictions iteratively by refining probability distributions across multiple decoder layers [157].
- Decoupled Distillation Focal Loss (DDF): DDF is a higher-level loss used for knowledge distillation, transferring information from deeper layers to shallower ones. It focuses on enhancing detector performance for hard-to-detect objects while improving overall detection [157].
- Bounding Box Loss (L1): L1 loss is a low-level loss that minimizes the spatial distance between predicted and true bounding box coordinates. It directly improves the accuracy of bounding box locations [158].
- Generalized Intersection over Union (GIoU) Loss: GIoU loss is a mid-level loss that improves localization accuracy by extending IoU to handle non-overlapping bounding boxes. It incorporates a distance penalty, refining bounding boxes even when they do not overlap [158].

In the default DEIMv2 model, applying the FGL and DDF losses to the ultra-lightweight models Pico, Femto, and Atto degraded their performance. This is primarily due to their limited capacity and relatively weaker baseline performance, which limits the effectiveness of self-distillation, where knowledge is transferred from the model's own predictions to enhance accuracy [159]. Consequently, the FGL and DDF components were excluded from training the ultra-lightweight variants to reduce additional complexity that could hinder their performance.

To determine which predicted bounding box corresponds to which ground truth, the Hungarian Matcher is used. It operates by minimizing a cost function, which takes into account three key factors: Classification, which checks whether the predicted box matches the ground truth class. Localization, which assesses where the box is placed. In additio to IoU/GIoU, which measures how well the predicted box overlaps with the ground truth box. However, in over 98% of the patches, there is only one whale, as the images are cropped to a size of 64 x 64. Therefore, these matching settings are not considered relevant for this purpose.

Data Augmentations

To enhance the generalization and robustness of the model, DEIMv2 incorporates several data augmentations. These augmentations are categorized based on their function and effects on the input data, as outlined in this section, with visual examples provided in Figure 5.52. The default DEIMv2 parameters for these augmentations are also detailed, but do not yet reflect the hyperparameter search settings.

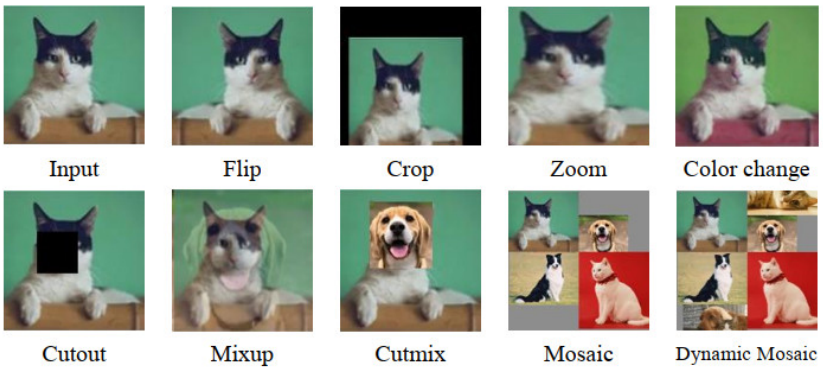


Figure 5.52: Examples of common data augmentation techniques, including geometric transformations, photometric changes, and sample-mixing methods, used to increase data diversity and improve model generalization. [160]

Geometric Transformations These transformations alter the spatial arrangement of the image and its objects. [161]

- Mosaic: This augmentation combines four images, resizes them, stitches them together, and then randomly cuts out a section to form the final image [162]. It increases diversity and improves model generalization. However, due to the small initial image size (64x64), cropping and upscaling might result in reduced spatial information, diminishing the model's ability to detect fine details.
- Random Zoom Out: This transformation zooms out the image, making the object smaller while adding a background, filled with a specified color. Similar to Mosaic, this augmentation can be less effective for small images like 64x64, as it further reduces object size. Nonetheless, it improves generalization to objects of different sizes.
- Random Horizontal Flip: The image is flipped horizontally, which helps the model learn invariant features. This transformation works well even for small images like 64x64, as it does not alter the object's size or position.

Photometric Distortions These augmentations modify the image's color properties to simulate various lighting conditions and contrast, which is beneficial for enhancing generalization. [161]

- Random Photometric Distort: Alters the brightness, contrast, and saturation of the image. This helps the model generalize across varying lighting conditions. It operates regardless of image size.

Bounding Box and Object Manipulations These transformations focus on ensuring that the model attends to the correct areas of the image and handles bounding boxes accurately. [161]

- Random IoU Crop: Crops the image so that the intersection-over-union (IoU) between the crop and the object of interest is maximized. For small images, this might not always be useful, as the crop could eliminate the object or make it less informative for training.
- Sanitize Bounding Boxes: Ensures the bounding boxes are valid after RandomIoUCrop, meaning not too small or outside the image. This transformation eliminates bounding box noise or errors.
- Convert Boxes: Converts bounding boxes from a format like top-left and bottom-right coordinates to center (cx, cy) and width, height (cxcywh). This transformation is essential for ensuring consistency in bounding box formats.

Image Resizing and Normalization These augmentations ensure that the input data is standardized and appropriately scaled for the model. Image resizing might be problematic for 64x64 images, as it could reduce the object resolution too much, especially for very small objects.

- Resize: Resizes the image to a fixed size. For small images, resizing typically involves upscaling. While this increases the object size in the image, it can also introduce padding to maintain the aspect ratio, leading to larger background areas. This padding could reduce the signal-to-noise ratio, making the background more prominent in the image. Additionally, resizing increases the computational cost and could introduce blur in the resized image [163].
- Normalize: Normalizes the pixel values of the image by subtracting the mean and dividing by the standard deviation for each channel (R, G, B). This ensures the model receives standardized scaled inputs.

Object-level Augmentations (O2O) These transformations involve adding, blending, or mixing objects within the images, introducing diversity and improving generalization. [161]

- Mixup: Mixes two images together by creating a convex combination. This helps the model learn from multiple examples at once, seeing multiple objects in one image. However, this might also reduce the visibility of individual objects [164].
- CopyBlend: Adds new objects without backgrounds into the image, and blends them so that their edges are preserved. This introduces new objects while maintaining high-quality boundaries, which enhances robustness [165]. Its effect on the current dataset remains uncertain. Because the dataset contains black background regions at high off-nadir angles, blending may artificially simplify the

detection task and therefore not always remain fully representative of real imagery. Restricting CopyBlend to the image region only falls outside the scope of this work. The optimal settings are therefore treated as hyperparameters to be explored experimentally.

The parameters used for the transformations are summarized in Table 5.10.

Table 5.10: Default Data Augmentation Parameters for DEIMv2-S

Augmentation	Parameter	Value
Mosaic	*output_size*	320
	rotation_range	10 (degrees)
	translation_range	[0.1, 0.1] (random translation in x, y directions)
	scaling_range	[0.5, 1.5] (scaling factor)
	probability	1.0 (100% chance)
	fill_value	0 (background filled with black)
Random Zoom Out	*fill*	0 (black background)
Random Photometric Distort	*p*	0.5 (50% chance)
Random IoU Crop	*p*	0.8 (80% chance)
Sanitize Bounding Boxes	*min_size*	1 (boxes smaller than 1 pixel are removed)
Convert Boxes	*fmt*	'cxcywh' (center x, center y, width, height)
	normalize	True (normalize to [0, 1])
Resize	*size*	[640, 640]
Normalize	*dtype*	'float32' (32-bit floating point)
	scale	True (pixel values scaled between 0 and 1)
	mean	[0.485, 0.456, 0.406]
	std	[0.229, 0.224, 0.225]
Mixup	*mixup_prob*	0.5 (50% chance)
	mixup_epochs	[4, 29]
CopyBlend	*copyblend_prob*	0.5 (50% chance)
	copyblend_epochs	[4, 120]

Scheduler

The following settings control the learning rate schedule and the application of augmentations during training.

- Total Epochs: The total number of training epochs is 132.
- Flat Epoch: The learning rate remains constant until epoch 64, after which it decays according to the flat cosine scheduler.
- No-Augmentation Epoch: The last 12 epochs, from epoch 121 to epoch 132, are dedicated to training the model on original, unaltered data. During this phase, no augmentations are applied, allowing the model to finalize its learning with the original images.
- Epoch-Specific Augmentations:
  - Mosaic: Applied from epochs 4 to 29. This augmentation combines four images into one, increasing data diversity.
  - Mixup: Applied from epochs 4 to 29. This technique blends two images together, improving generalization.
  - CopyBlend: Applied from epochs 4 to 120. This adds new objects into the image, enhancing feature learning.
- Learning Rate: The learning rate controls how much the model's weights are adjusted with each step during training, affecting the speed of learning.

- Warmup Iterations: The learning rate gradually increases from a lower value to the initial learning rate over the first 2000 iterations. This stabilizes training during the early phases.
- Learning Rate Scheduler: The learning rate remains flat during the first 64 epochs, before the flat epoch, and then decays following a cosine schedule at rate *lr_gamma*. This ensures that training starts with a stable learning rate and gradually reduces it for fine-tuning.
- Weight Decay: A regularization technique that adds a penalty to large weights, to prevent overfitting and improve generalization.
- AdamW: An adaptive learning rate optimizer that combines the benefits of Adam, for adjusting learning rates based the first and second moments of the gradients, and weight decay to regularize the model by penalizing large weights, which improves convergence and prevents overfitting.
- Batch Size: The number of training examples used in one forward and backward pass of the model. A larger batch size can speed up training and provide a more stable gradient estimate, while a smaller batch size might result in more noise in the gradients and a longer training time, but saving compute, allows the model to generalize better.

The parameters for the DEIMv2-S scheduler and related settings are summarized in Table 5.11 [146].

Table 5.11: Default Scheduler Parameters for DEIMv2-S [146].

Parameter	Value
epoches	132
flat_epoch	64
no_aug_epoch	12
optimizer	AdamW
warmup_iter	2000
lr	5e-4
lrsheduler	flatcosine
lr_gamma	0.5
batch_size	32

Baseline Changes

The following changes were immediately applied based on the baseline settings.

The original models use an input size of 640 x 640, whereas the patch images are 64 x 64. To minimize the processing power requirements from the outset while preserving the size of the internal layers after downsampling, the input size was reduced to 320 x 320. This adjustment also impacted parameters related to the input size, including `eval_spatial_size`, Mosaic `output_size`, and resizing operations, which were modified following the repository guidelines for consistency [146]. The effect of resolution input is to be verified in subsection 5.4.3.

Normalization was then applied to match the mean and standard deviation values from the training dataset. In the case of cross-validation, the mean and standard deviation were computed over the training folds only, and the values for all RGB channels were automatically replaced in the configuration files.

During a trial run, it was observed that the logs output only the training loss, and not the validation loss. To examine overfitting as described in subsection 5.4.3, the configuration was adjusted to evaluate both the training and validation loss. Also, the option to monitor progress using TensorBoard was enabled.

Evaluation plots were created manually by parsing the training logs to track AP and AR scores over the epochs, which allowed an assessment of the model's convergence and stability. Additionally, training and validation loss curves were analyzed to identify potential overfitting or underfitting. A CSV file was automatically generated to summarize the performance metrics to provide a detailed comparison across different configurations.

In addition, it was decided to evaluate performance based on IoU 0.50, and therefore access AP_{50}. This threshold is selected because the whale annotations are manually derived from VHR imagery and may contain minor spatial inaccuracies. Moreover, whales occupy only a small number of pixels,

making IoU highly sensitive to even a one-pixel offset. Since IoU is not scale-aware, this effect becomes stronger for smaller whales, which occur more frequently at higher off-nadir angles due to geometric projection. Evaluating across stricter thresholds up to 0.95 would therefore disproportionately penalize small localization differences and reduce comparability across viewing angles. Because the objective is to quantify detection robustness under off-nadir distortions rather than sub-pixel bounding-box refinement, AP_{50} provides a stable and practically-relevant evaluation criterion.

5.4.3. Hyperparameter Optimization

Given that the input size images are originally 64x64, some augmentations might be less beneficial or even counterproductive, as highlighted in Equation 5.4.2. Therefore, given the default settings in Equation 5.4.2, the effect of several hyperparameters was examined to optimize the detection model. Specifically, while the original input patches are small, 64 x 64, it was mainly aimed at verifying the hypothesis of the disadvantageous effects of Mosaic, RandomIoUCrop, after ensuring convergence within a fixed batch size, and number of epochs, and input dimension to optimize training, and for input dimension also operational efficiency, and learning rate to ensure stability and progress. In addition to scheduler settings, aiming to obtain a detection model that is both accurate and computationally efficient.

The optimization was organized in multiple stages, where early stages were used for exploration and scheduler settings, and in later stages, small refinements will be made after fixing scheduler settings.

The search itself was automated through a Python configuration generator. Starting from a default DEIMv2 configuration file, the script generates modified YAML files by replacing a predefined set of hyperparameters. Each generated configuration corresponds to one unique combination of selected values. The script enforces balanced marginal counts over the search dimensions. This means that each candidate value occurs approximately equally often across the generated set, while still combining them randomly into unique tuples. In this way, the search avoides that certain values are underrepresented by chance.

To prevent duplicate experiments in the case of random search, the script enforces uniqueness over the generated oonfigs. In addition, balanced target counts are imposed for each individual search dimension. This means that for a generated set of N configurations, each candidate value appears approximately N/m times, where m is the number of possible values for that variable. The resulting search can then be interpreted as a balanced random search over unique tuples, instead of a purely independent random draw, with the exception for settings that modify training duration, like batch size, if they give the possiblity for more runs.

Hyperparameter Dependencies

To remain consistent with the repository implementation, several hyperparameter dependencies were taken into account [146]. Specifically, batch size and learning rate are coupled through a linear scaling rule, so that changes in batch size are accompanied by corresponding changes in the effective optimizer settings. Let the default batch size be $B_0 = 32$ and the sampled batch size be B. The batch-scaling factor is then defined by Equation 5.80,

$$f = \frac{B}{B_0}. \tag{5.80}$$

Given a sampled baseline learning-rate choice η_{choice}, the effective optimizer base learning rate is obtained from the repository scaling rule in Equation 5.81,

$$\eta = \eta_{\text{choice}}\, f. \tag{5.81}$$

For the generated repository configuration used in this work, the learning rates of the first two optimizer parameter groups, corresponding to the DINOv3 backbone parameters, are then set according to Equation 5.82,

$$\eta_{\text{backbone}} = \frac{\eta}{10}. \tag{5.82}$$

Thus, the sampled baseline learning rate and the sampled batch size together determine both the effective optimizer base learning rate and the corresponding backbone learning rate.

Similarly, the Exponential Moving Average (EMA) and warmup settings are scaled with batch size. Let d_0 denote the default EMA decay. The adjusted EMA decay is given by Equation 5.83,

$$d = 1 - (1 - d_0) f, \tag{5.83}$$

while the EMA warmup steps and learning-rate warmup duration are adapted according to Equation 5.84 and Equation 5.85,

$$N_{\text{EMA,warmup}} = \text{round}\left(\frac{1000}{f}\right), \tag{5.84}$$

$$N_{\text{LR,warmup}} = \text{round}\left(\frac{500}{f}\right). \tag{5.85}$$

The sampled input resolution also propagates to multiple locations in the configuration. If the selected resolution is denoted by $[R, R]$, the resize operations in both the training and validation pipelines are replaced by $[R, R]$, the collate-function base size is set to R, and the Mosaic output size is updated to $R/2$. Consequently, the search dimension labelled as resolution affects both the final detector input size and the spatial scale at which the Mosaic augmentation is constructed.

DEIM schedule relations In addition to the hyperparameters varied by the search script, the DEIM training schedule contains several epoch boundaries that are coupled through the scheduling rules adopted from the DEIM repository. Several are derived from the total number of epochs and the model-size parameter n.

For the default model used in this thesis, the no-augmentation phase length is given by

$$\texttt{no\_aug\_epoch} = 4n. \tag{5.86}$$

Since the default configuration uses $\texttt{no\_aug\_epoch} = 12$, it follows that

$$n = 3. \tag{5.87}$$

With n fixed by the selected model size, the no-augmentation length remains fixed unless the repository rule is intentionally broken for exploratory studies. The remaining schedule boundaries are then derived from the total number of epochs.

Let the total number of epochs be denoted by E and define the end of the augmented phase as

$$E_{\text{aug}} = E - \texttt{no\_aug\_epoch}. \tag{5.88}$$

The DEIM repository then defines the flat epoch by

$$\texttt{flat\_epoch} = 4 + \left\lfloor \frac{E_{\text{aug}}}{2} \right\rfloor. \tag{5.89}$$

The corresponding policy and collate-function epoch ranges are derived directly from this quantity. These relations are summarized in Table 5.12.

As an example, the original starting schedule in this work used

$$E = 132, \qquad \texttt{no\_aug\_epoch} = 12. \tag{5.90}$$

This gives

$$E_{\text{aug}} = 132 - 12 = 120, \tag{5.91}$$

and therefore

$$\texttt{flat\_epoch} = 4 + \left\lfloor \frac{120}{2} \right\rfloor = 64. \tag{5.92}$$

The corresponding policy boundaries become $[4, 64, 120]$, the mixup range becomes $[4, 64]$, the copyblend range becomes $[4, 120]$, and the collate stop epoch becomes 120. The matcher change epoch was set near the end of the augmented phase, following the repository rule [146].

Table 5.12: DEIM repository schedule relations used to derive epoch-dependent training settings.

Setting	Definition
No-augmentation phase length	$no\_aug\_epoch = 4n$
End of augmented training phase	$E_{\text{aug}} = E - no\_aug\_epoch$
Flat epoch	$flat\_epoch = 4 + \left\lfloor \frac{E_{\text{aug}}}{2} \right\rfloor$
Policy schedule epochs	$policy.epoch = [4,\ flat\_epoch,\ E_{\text{aug}}]$
Mixup active range	$mixup\_epochs = [4,\ flat\_epoch]$
CopyBlend active range	$copyblend\_epochs = [4,\ E_{\text{aug}}]$
Collate stop epoch	$stop\_epoch = E_{\text{aug}}$
Matcher update epoch	$matcher\_change\_epoch \approx 0.9\,E_{\text{aug}}$

Cross-Validation Strategy

An overview of the staged hyperparameter optimization procedure is given in Table 5.13. The full procedure was split into six stages, where the first stage was used for rapid screening, middle stages for scheduler and learning rate settings to ensure optimal convergence, and later stages for augmentation parameter optimization.

Table 5.13: Overview of the staged hyperparameter optimization procedure.

Stage	Purpose	Configurations / runs	Folds	Output
1	Exploration	10 configurations	1	Initial screening of the search space and identification of promising regions for learning rate, batch size, augmentation strength, and input resolution.
2	Learning Rate Selection	6 learning rates	4	Selection of the learning rate for cross-fold optimization, balancing convergence speed and training stability.
3	Schedule Refinement	9 schedules	4	Comparison of total epoch counts and no-augmentation lengths, leading to a shortened cross-validation schedule for later augmentation experiments.
4	Augmentation Global Search	25 random-search configurations + 1 zero-augmentation baseline	4	Identification of promising parameter ranges for mosaic probability, RandomIoUCrop, mixup, and CopyBlend.
5	Augmentation Local Search	8 configurations	4	Final cross-fold refinement of the narrowed augmentation search space and selection of the best hyperparameter configuration.
6	Final Deployment training	2 epoch settings	Pooled non-test data (95/5 split)	Final model trained on two epochs, on the seven non-test locations with a 95%/5% training-validation split.

The dataset was split into four cross-validation folds. Given a total of 9 locations, with 2 reserved for the test set, the total number of possible validation-set combinations from the remaining 7 locations would be $\binom{7}{2} = 21$. However, due to computational constraints and limited time budgets, the fold design was reduced to a smaller set while ensuring that each location appeared at least once in the validation set. The final split is given in Table 5.14. The datasets for all locations are visualized in Appendix D.

Pelagos2016 and Auckland2006 were selected for the test set. These locations represent the two source-resolution as described in subsection 5.4.1, with Pelagos2016 corresponding to the 0.30 m ground sampling distance group and Auckland2006 to the 0.54 m ground sampling distance group. In addition, Pelagos2016 is the only location containing fin whales of approximately 20-22 m, whereas the remaining locations mainly contain whales in the range of 13-15 m. Together, these two locations therefore represent two distinct extremes within the dataset: a higher-resolution case with larger whales, and a lower-resolution case with coarser spatial detail. This makes them suitable test locations for evaluating how well the trained model generalizes to unseen data that differs from the training set, similar to real operational scenarios.

The two locations are also distinct in terms of illumination conditions and whale shape. Pelagos2016 contains a larger range of radiometric conditions, including variation in brightness and noise level, whereas Auckland2006 contains a broader range of whale shapes and poses. Therefore, selecting these two locations increases both radiometric diversity and geometric diversity within the test set, making the evaluation on unseen locations more representative.

The cross-validation folds were generated with a fixed random seed. First, all possible validation-location combinations of size two were enumerated from the seven remaining locations. The folds were then selected greedily, one by one, by choosing at each step the remaining validation combination that kept the number of validation appearances per location as balanced as possible. This was done by minimizing first the difference between the most and least frequently used validation locations, then the highest usage count, and finally the variance across these counts. The random seed was only used to break ties between equally scoring candidates.

After randomly generating the folds, their composition was interpreted in terms of data representation. The first fold contains Auckland2011 and Witsand2009, corresponding to WorldView-2 and GeoEye-1 data respectively, both belonging to the higher ground sampling distance group. The second fold contains Ignacio2017 and Maui2015, both captured with WorldView-3 and therefore belonging to the lower ground sampling distance group.

In folds 3 and 4, the Valdes locations were, by coincidence, kept together. Valdes2014 contains WorldView-3 data, whereas Valdes2012 and Valdes2016 contain WorldView-2 data. In practice, this also allows the training-validation loop to better reflect an operational scenario in which the satellite is applied to previously unseen locations. At the same time, although these samples originate from the same region, the acquisition conditions still vary substantially between years. For these reasons, the folds were not regenerated.

In addition, because one location had to appear twice in the validation set, Valdes 2016 was selected by the seeded split generation. As discussed in section 6.2, the Valdes 2016 data is solely affected strongly by the absence of input preprocessing. Although this makes it the least representative location in the dataset, where future work should include this normalization step, retaining it in the validation set provides a more informative estimate of how well the model generalizes to uncommon scenes.

Since the final training is performed on 95% of the available training data, with all locations combined, the final model is expected to generalize better than the models evaluated during cross-validation on the test set.

Stage 1: Exploration. The first stage was designed to obtain an initial indication of how sensitive the model is to the selected hyperparameters. Since multiple runs could be executed in parallel, this exploratory stage was carried out using ten configurations on the first cross-validation fold. In this way, a relatively broad part of the search space could be screened at low computational cost before committing to more extensive cross-fold experiments.

The purpose of this stage was to determine whether the selected search dimensions produce different model behaviour and to identify promising regions of the search space for subsequent experiments. Because this stage functioned as an initial pilot, it was restricted to a single fold in order to keep evaluation time manageable.

Table 5.14: Dataset split used for the four-fold cross-validation. Each fold defines separate training, validation, and test location sets.

Fold	Train locations	Validation locations	Test locations
Fold 1	Ignacio2017, Maui2015 Valdes2012, Valdes2014 Valdes2016	Auckland2011 Witsand2009	Pelagos2016 Auckland2006
Fold 2	Auckland2011, Valdes2012 Valdes2014, Valdes2016 Witsand2009	Ignacio2017 Maui2015	Pelagos2016 Auckland2006
Fold 3	Auckland2011, Ignacio2017 Maui2015, Valdes2012 Witsand2009	Valdes2014 Valdes2016	Pelagos2016 Auckland2006
Fold 4	Auckland2011, Ignacio2017 Maui2015, Valdes2014 Witsand2009	Valdes2012 Valdes2016	Pelagos2016 Auckland2006

The search space was defined through a random search over the following parameters:

$$\text{Learning rate} \in \{10^{-3}, 10^{-4}, 10^{-5}\},$$

$$\text{Batch size} \in \{32, 16, 8\},$$

$$\text{RandomIoUCrop probability} \in \{0.8, 0.4, 0.0\},$$

$$\text{Mosaic probability} \in \{0.8, 0.4, 0.0\},$$

$$\text{Resolution} \in \{320, 256\}.$$

A resolution of 128 had also been considered in preliminary experiments. However, this setting was not compatible with the default number of object queries, which remained fixed at 300 in the repository configuration. For a 128×128 input image, this produced an unfavourable detector configuration relative to the image size. Although the number of queries could in principle have been reduced, this issue was identified only after the runs had already been completed. Repeating those experiments would have reduced comparability with the remaining configurations and would have shifted effort away from the main optimization study. Therefore, the exploration was restricted to resolutions of 256 and 320.

Since this stage consisted of only ten runs on a single fold, the resulting trends should be interpreted cautiously. The goal at this point was to determine whether clear sensitivities were already visible and whether certain parameter regions appeared unpromising. For that reason, the results are first summarized through grouped tables, after which selected loss curves are inspected in more detail.

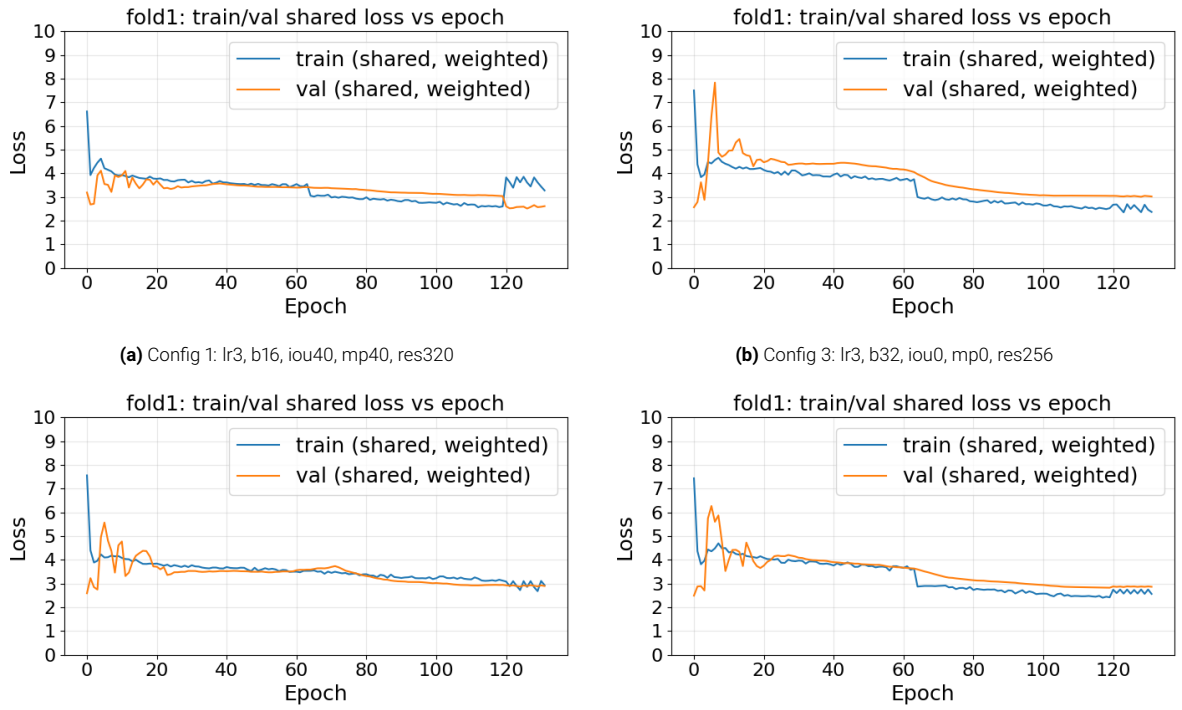
The grouped AP_{50} results for learning rate are given in Table 5.15. The average differences between the tested values remain small, on the order of approximately 0.01. At this stage, this can be interpreted in two ways. On the one hand, it may indicate that the model converges robustly across a relatively broad learning-rate range. On the other hand, because multiple hyperparameters vary simultaneously and the analysis is limited to a single fold with ten configurations, the observed AP differences cannot be isolated to individual parameters, and interaction effects likely dominate the results.

A secondary explanation is that the task itself is relatively simple, with one object per image and limited visual variability, which can reduce sensitivity to hyperparameter choices. However, this cannot be concluded from this stage alone and is further investigated in the cross-validation experiments.

Table 5.15: Average AP_{50} grouped by learning rate.

Learning Rate	Runs	Average AP_{50}
3×10^{-4}	1, 3, 8, 9	0.769
4×10^{-4}	2, 5, 6, 7	0.764
5×10^{-4}	4, 10	0.758

To analyze the effect of higher learning rates on convergence behaviour, the four configurations with a learning rate of $1e{-}3$ were examined in detail. The corresponding training and validation loss curves are shown in Figure 5.53.



(a) Config 1: lr3, b16, iou40, mp40, res320

(b) Config 3: lr3, b32, iou0, mp0, res256

(c) Config 8: lr3, b32, iou80, mp80, res256

(d) Config 9: lr3, b32, iou0, mp0, res320

Figure 5.53: Training and validation loss curves for four hyperparameter configurations explored in fold 1 for learning rate 1e-3.

After epoch 4, the augmentation pipeline is activated, including Mosaic, MixUp, and CopyBlend. This transition introduces a sharp increase in both training and validation loss across all configurations in Figure 5.53, reflecting the shift from original to augmented data.

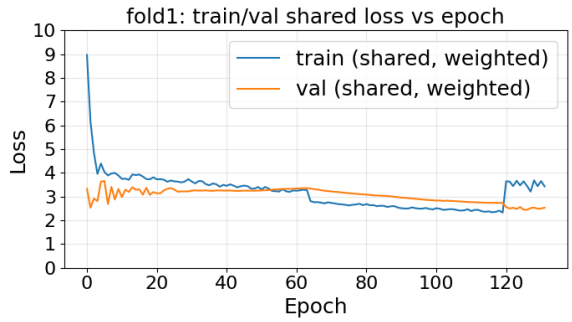
Between approximately epochs 10 and 60, both training and validation loss decrease gradually. This indicates that the model adapts to the augmented data distribution, although the rate of improvement remains limited.

At the flat epoch. around epoch 64, where the learning rate begins to decay and augmentation strength is reduced, both training and validation loss decrease again. This consistent drop across configurations indicates that the initial learning rate mainly supports coarse optimization, while the decay phase enables further refinement.

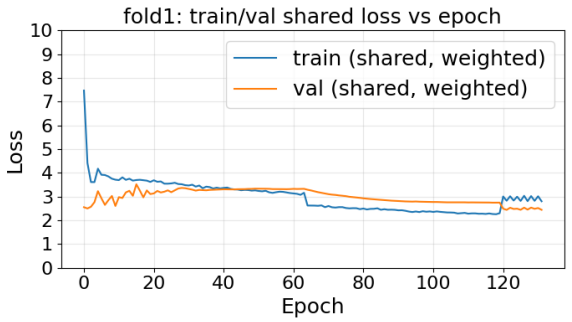
After epoch 120, where CopyBlend is disabled, a divergence between training and validation loss is observed. The training loss increases, while the validation loss remains stable or decreases slightly. This indicates that CopyBlend introduces samples that are easier to fit during training but are not representative of the validation data.

For all configurations, the validation loss flattens around epoch 60-80, while the training loss continues to decrease. This indicates that further training primarily improves performance on the training set without translating to validation improvements, in other words, overfitting occurs beyond this point.

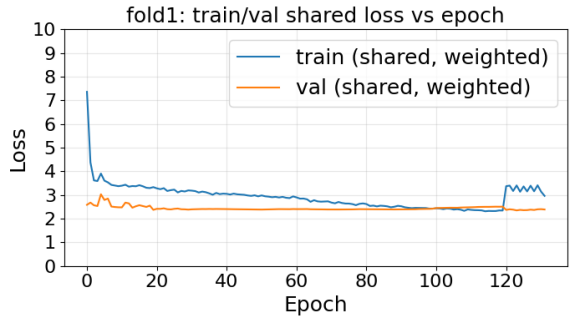
Finally, the validation loss remains close to or slightly below the training loss. This indicates that the validation data in this fold is of comparable difficulty to the training data. In some cases, the validation loss drops below the training loss, which can be attributed to the absence of augmentations in the validation pipeline, while the training data includes additional augmented samples that are more difficult to fit. The absolute loss values are scaled according to Equation 5.79, and are therefore not directly interpretable in magnitude.



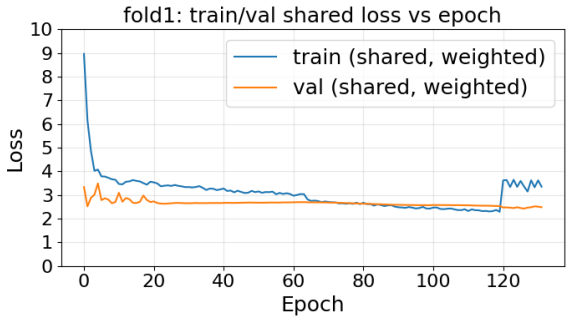
(a) Config 2: lr4, b32, iou80, mp0, res320



(b) Config 5: lr4, b8, iou40, mp0, res256



(c) Config 6: lr4, b8, iou80, mp80, res320

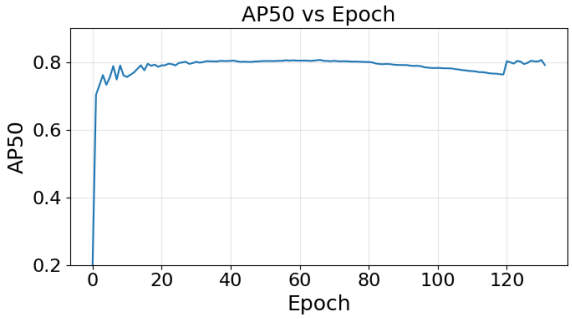


(d) Config 7: lr4, b32, iou80, mp40, res320

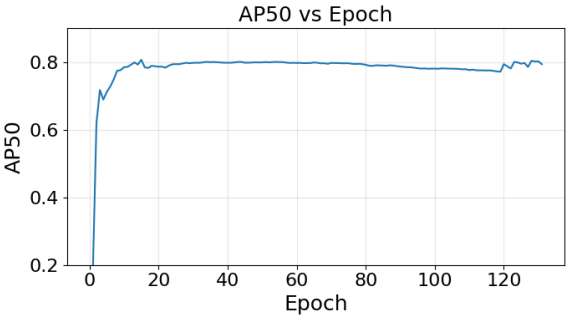
Figure 5.54: Training and validation loss curves for four hyperparameter configurations explored in fold 1 for learning rate 1e-4.

For learning rate $1e{-}4$, the initial augmentation phase (epoch 4-29) appears more stable compared to the higher learning rate, although small fluctuations remain. The validation loss flattens early, and for configurations with non-zero mosaic probability it remains nearly constant after approximately epoch 30, while the training loss continues to decrease. This indicates earlier overfitting compared to the higher learning rate setting.

This behaviour is confirmed by the AP_{50} curves in Figure 5.55, where performance peaks around epoch 30-40 and subsequently declines for these configurations. After copyblend is disabled, the AP recovers, indicating that the model had partially adapted to augmented samples that are not representative of the validation data.



(a) Config 6: lr4, b8, iou80, mp80, res320

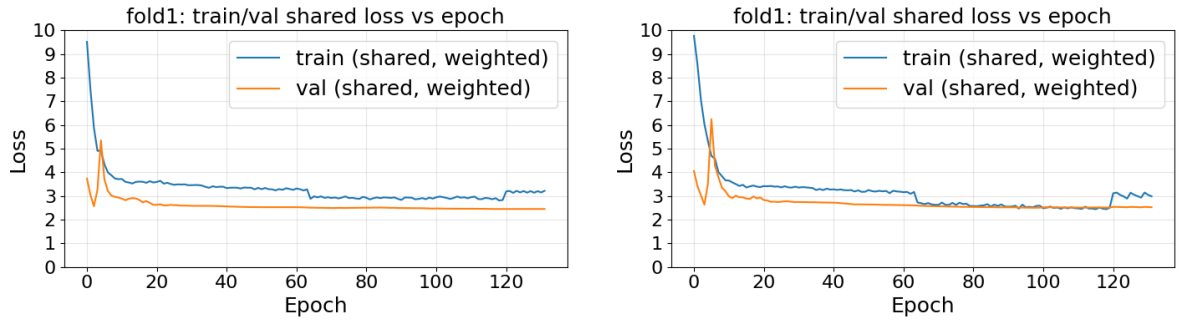


(b) Config 7: lr4, b32, iou80, mp40, res320

Figure 5.55: Validation AP_{50} over training epochs for two hyperparameter configurations explored in fold 1 for learning rate 1e-4 and non-zero mosaic probability scores. The curves show overfitting after approximately epoch 75.

This behaviour suggests that, under strong augmentation settings, the model may partially rely on features introduced by CopyBlend that are not present in the validation data. As a result, performance decreases during training and recovers once CopyBlend is disabled. This effect is further investigated in later stages.

The loss curves for learning rate $1e{-5}$ are shown in Figure 5.56.



(a) Config 4: lr5, b16, iou40, mp80, res256

(b) Config 10: lr5, b32, iou0, mp80, res256

Figure 5.56: Shared training and validation loss curves over epochs for two hyperparameter configurations explored in fold 1 for learning rate 1e-5.

For learning rate $1e{-5}$, the loss curves in Figure 5.56 show a more stable training process. As in the previous settings, the activation of augmentations at epoch 4 introduces an initial increase in validation loss. However, both training and validation loss decrease steadily afterwards.

At epoch 64, where the learning rate and augmentation strength begins to decay, the training loss decreases further, while the validation loss remains largely unchanged. This indicates that the model has already reached a stable regime before the decay phase, and additional training mainly refines the training loss without improving validation performance.

In both configurations, with relatively low RandomIoUCrop and high mosaic probability, the curves converge closely. After copyblend is disabled, the training loss increases, consistent with the removal of augmented samples that are easier to fit. Compared to higher learning rates, no strong divergence between training and validation loss is observed, indicating reduced overfitting.

Overall, learning rate $1e{-5}$ shows the most stable behaviour among the tested configurations. However, only two runs were completed for this setting, both under similar augmentation conditions and on a single fold. Therefore, this observation is not sufficient to fix the learning rate, and it is further evaluated in the next stage.

Next, the effect of batch size was evaluated. As shown in Table 5.16, the differences in AP_{50} between batch sizes are small, with batch size 32 showing a marginally higher average performance. In addition, batch size 32 significantly reduces training time compared to smaller batch sizes. Therefore, batch size 32 was selected for the subsequent experiments.

Table 5.16: Average AP_{50} grouped by batch size.

Batch Size	Runs	Average AP_{50}
32	2, 3, 7, 8, 9, 10	0.770
16	1, 4	0.753
8	5, 6	0.763

The effects of RandomIoUCrop and mosaic probability are summarized in Table 5.17 and Table 5.18. In both cases, the highest average AP_{50} is obtained when the augmentation is disabled, with performance decreasing for higher values. This suggests that strong spatial augmentations can remove or distort relevant object information, particularly at low input resolutions.

However, the observed differences are small and based on a limited number of runs on a single fold, and no consistent monotonic trend is present across all configurations. Therefore, these results are not sufficient to justify disabling either augmentation entirely at this stage.

Instead, both parameters are set to intermediate values to retain their effect while targetting the search towards the more promising range. Consequently, RandomIoUCrop and mosaic probability are both set to 0.2 for the subsequent experiments.

Table 5.17: Average AP_{50} grouped by IoU parameter.

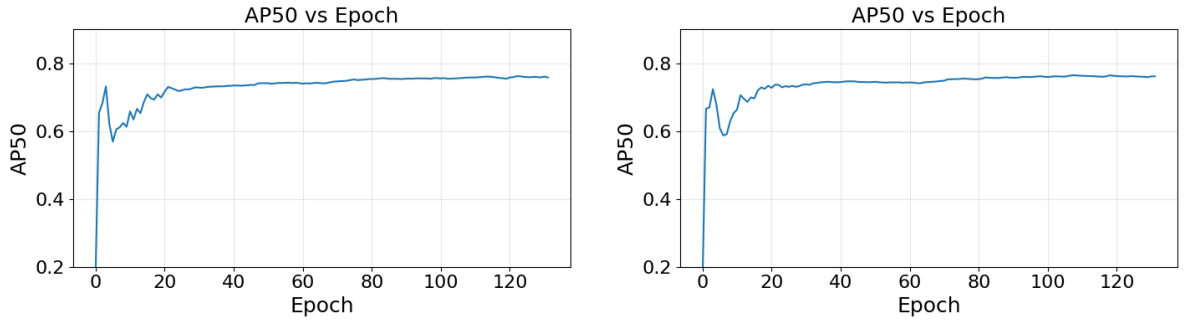
IoU Parameter	Runs	Average AP_{50}
0	3, 9, 10	0.774
40	1, 4, 5	0.758
80	2, 6, 7, 8	0.763

Table 5.18: Average AP_{50} grouped by MP parameter.

MP Parameter	Runs	Average AP_{50}
0	2, 3, 5, 9	0.775
40	1, 7	0.761
80	4, 6, 8, 10	0.758

Lastly, the effect of input resolution was evaluated. As shown in Table 5.19, a resolution of 320 achieves slightly higher AP_{50} compared to 256, with a difference of approximately 0.004. Given that the original input images are limited to 64×64, this marginal improvement is expected, as the upsampling does not introduce additional information but only refines spatial representation.

A direct comparison between configurations 3 and 9, shown in Figure 5.57, confirms that both resolutions follow nearly identical AP_{50} trends over training. While resolution 256 shows slightly higher fluctuations in early epochs, the overall convergence behaviour and final performance remain effectively the same.



(a) Config 3: lr3, b32, iou0, mp0, res256

(b) Config 9: lr3, b32, iou0, mp0, res320

Figure 5.57: Validation AP_{50} over training epochs for two hyperparameter configurations explored in fold 1 for learning rate 1e-3 and zero mosaic probability.

It should be noted that this comparison is based on configurations without RandomIoUCrop and mosaic. While resolution effects may be more pronounced under stronger spatial augmentations, the remaining runs indicate that the overall performance difference remains small across configurations.

In addition to accuracy, resolution directly impacts computational cost. As shown in Table 5.19, reducing the input size from 320×320 to 256×256 decreases latency by approximately 10–15%. This reduction also translates to faster training, enabling a larger number of configurations to be evaluated within the same computational budget.

Given the marginal performance difference and the consistent reduction in computational cost, a resolution of 256×256 was selected for subsequent experiments.

Table 5.19: Average AP_{50} and average latency per image grouped by input resolution.

Resolution	Runs	Average AP_{50}	Average latency [ms/image]
256	3, 4, 5, 8, 10	0.764	21.38
320	1, 2, 6, 7, 9	0.768	24.61

Across all configurations in this stage, performance differences remain small on the single fold. This indicates that the current setup does not clearly separate hyperparameter effects under these conditions. Therefore, the next stage extends the evaluation to multiple folds to assess whether these trends persist under varying data distributions.

Stage 2: Learning Rate Selection. The objective of this stage is to determine an appropriate learning rate for subsequent experiments. While a learning rate of $1e{-}5$ appeared stable in Stage 1, this observation was based on a single fold and does not guarantee consistent performance across different data splits or more challenging configurations. Therefore, a dedicated learning rate sweep was conducted using four cross-validation folds.

To efficiently explore the parameter space within available computational resources, six configurations were evaluated in parallel, as summarized in Table 5.20. The learning rate was varied on a logarithmic grid with half-decade spacing, while all other parameters were kept fixed.

Based on observations from Stage 1, where training for 132 epochs showed diminishing returns, the number of epochs was reduced to 80. Performance was observed to plateau around epoch 64, and a small margin was added to ensure convergence. This choice prioritizes faster experimentation, while further refinement of the training schedule will be addressed in the next stage.

Table 5.20: Overview of the learning-rate-focused hyperparameter optimization runs. In this sweep, the batch size, augmentation probabilities, and input resolution were kept fixed, while only the learning rate was varied on a logarithmic grid with half-decade spacing.

Run	LR tag	Learning rate	Backbone LR	Batch size	RandomIoUCrop p	Mosaic prob.	Resolution
1	e40	10^{-4}	10^{-5}	32	0.2	0.2	256
2	e60	10^{-6}	10^{-7}	32	0.2	0.2	256
3	e50	10^{-5}	10^{-6}	32	0.2	0.2	256
4	e35	$10^{-3.5}$	$10^{-4.5}$	32	0.2	0.2	256
5	e55	$10^{-5.5}$	$10^{-6.5}$	32	0.2	0.2	256
6	e45	$10^{-4.5}$	$10^{-5.5}$	32	0.2	0.2	256
7	e30	10^{-3}	10^{-4}	32	0.2	0.2	256

The results are shown in Figure 5.58.

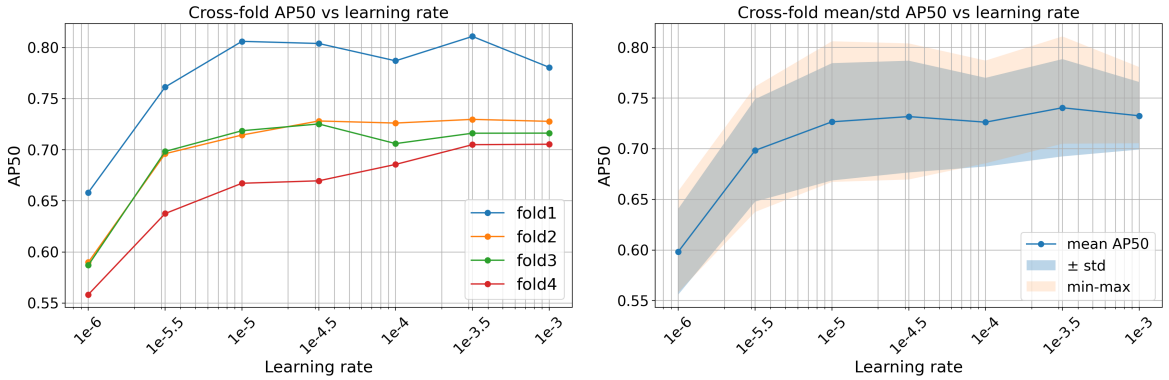


Figure 5.58: Cross-validation AP_{50} as a function of learning rate. Left: per-fold performance. Right: mean AP_{50} with variability across folds (standard deviation and range).

The training and validation loss curves for $1e{-}4$, $1e{-}4.5$, and $1e{-}5$ are compared to assess training stability,

as shown in Figure 5.60, Figure 5.61, and Figure 5.62. Additionally, Figure 5.59 is included for completeness, while Figure 5.63 illustrates the behavior when the learning rate becomes too small.

Figure 5.59: Training and validation loss curves for a learning rate of $10^{-3.5}$ across all folds. The model converges quickly but shows a consistently large gap between training and validation loss, indicating early overfitting in fold 3 and 4.

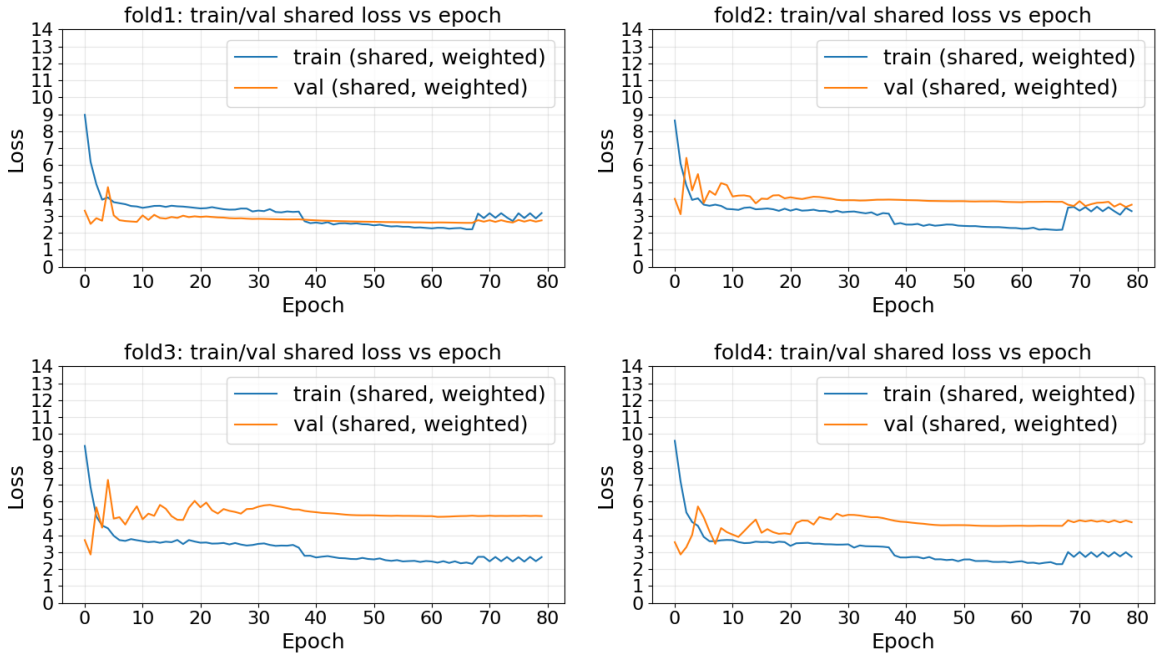


Figure 5.60: Training and validation loss curves for a learning rate of 10^{-4} across all folds. The model exhibits fast and stable convergence with a relatively small gap between training and validation loss for folds 1 and 2, indicating effective optimization and good generalization.

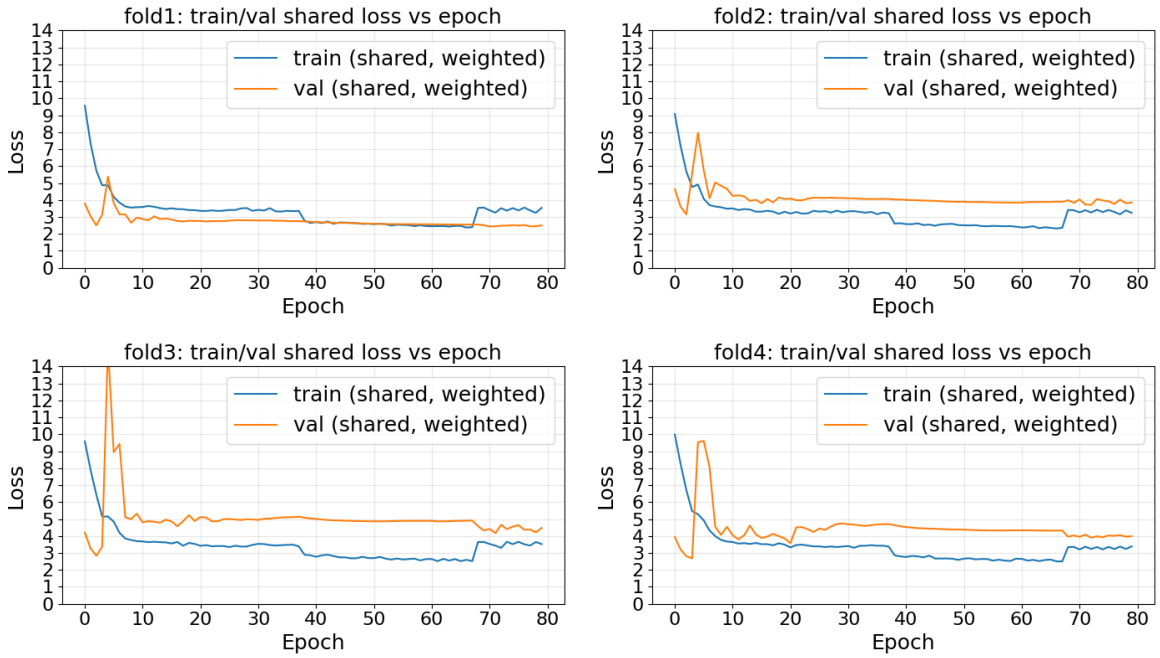


Figure 5.61: Training and validation loss curves for a learning rate of $10^{-4.5}$ across all folds. Convergence is stable but slower than for 10^{-4}.

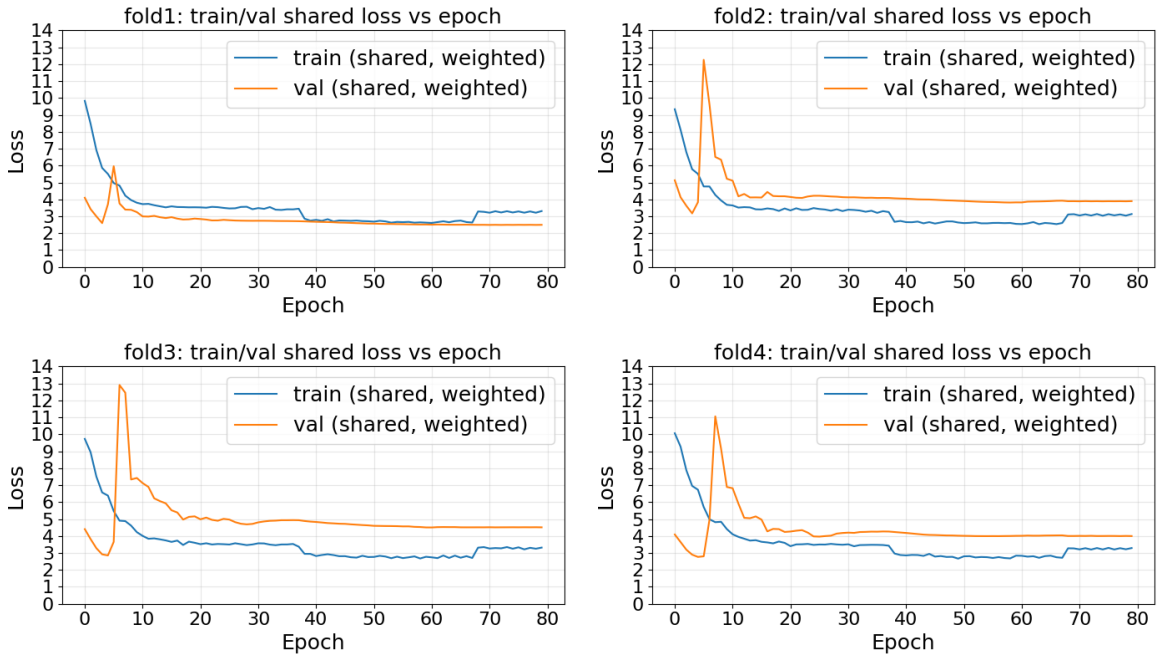


Figure 5.62: Training and validation loss curves for a learning rate of 10^{-5} across all folds. Optimization is slower and less effective, with higher validation loss peaks before learning how the augmentatations are applied.

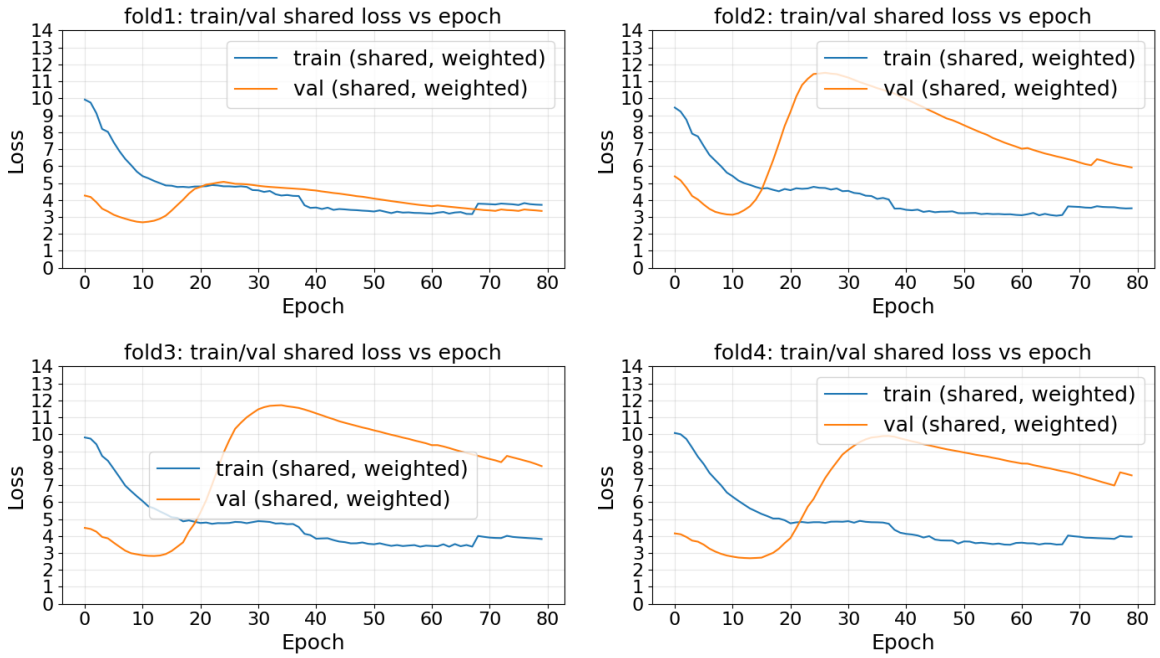


Figure 5.63: Training and validation loss curves for a learning rate of 10^{-6} across all folds. The model fails to converge properly within the given amount of epochs, where the validation loss peak is delayed, showing that the learning rate is too small for effective training.

At first, it can be observed that performance varies noticeably across folds. Fold 1 shows relatively well-aligned training and validation loss, with even slightly lower validation loss in some phases, indicating that the validation data may be easier while still remaining comparable. Fold 2 shows a moderate gap, whereas folds 3 and 4 exhibit a significantly larger divergence between training and validation loss, indicating stronger overfitting or a generalization gap.

This behavior is consistent with the dataset characteristics. Specifically, the Valdes 2016 dataset introduces a domain shift due to its higher original input values, given lack of input pre-processing, as discussed in subsection 6.2.7. Since this dataset is not sufficiently represented in the training splits of certain folds, the model has limited ability to generalize, leading to higher validation loss.

Comparing the different learning rates, $1e{-}5$ results in smoother validation loss curves than $1e{-}4$, indicating more conservative optimization. In contrast, $1e{-}4$ shows a steeper initial decrease in training loss, reflecting faster learning. While training loss remains stable across all configurations, $1e{-}5$ reaches a higher validation loss peak after the first few epochs.

This effect can be attributed to the delayed learning progress at lower learning rates. Since augmentations Mosaic and RandomIoUCrop are activated after the initial four epochs, the model trained with $1e{-}5$ has not yet sufficiently adapted to the data distribution, resulting in a temporary degradation in validation performance. Extending the initial no-augmentation phase could mitigate this effect.

The intermediate learning rate $1e{-}4.5$ shows behavior between the two extremes, but performs slightly worse on folds 1 and 3. The higher learning rate $1e{-}3.5$ achieves the highest mean AP_{50}, but exhibits rapid overfitting. In particular, validation performance stabilizes early while training loss continues to decrease, indicating reduced generalization capacity.

Although $1e{-}3.5$ shows strong AP_{50} performance, its early convergence makes it less suitable for subsequent hyperparameter optimization stages, where sensitivity to configuration changes during training is required. Therefore, it is excluded from further consideration.

Based on these observations, the learning rates $1e{-}4$, $1e{-}4.5$, and $1e{-}5$ were selected for further comparison. To distinguish between them, the corresponding AP_{50} curves over training epochs were analyzed, as shown in Figure 5.64.

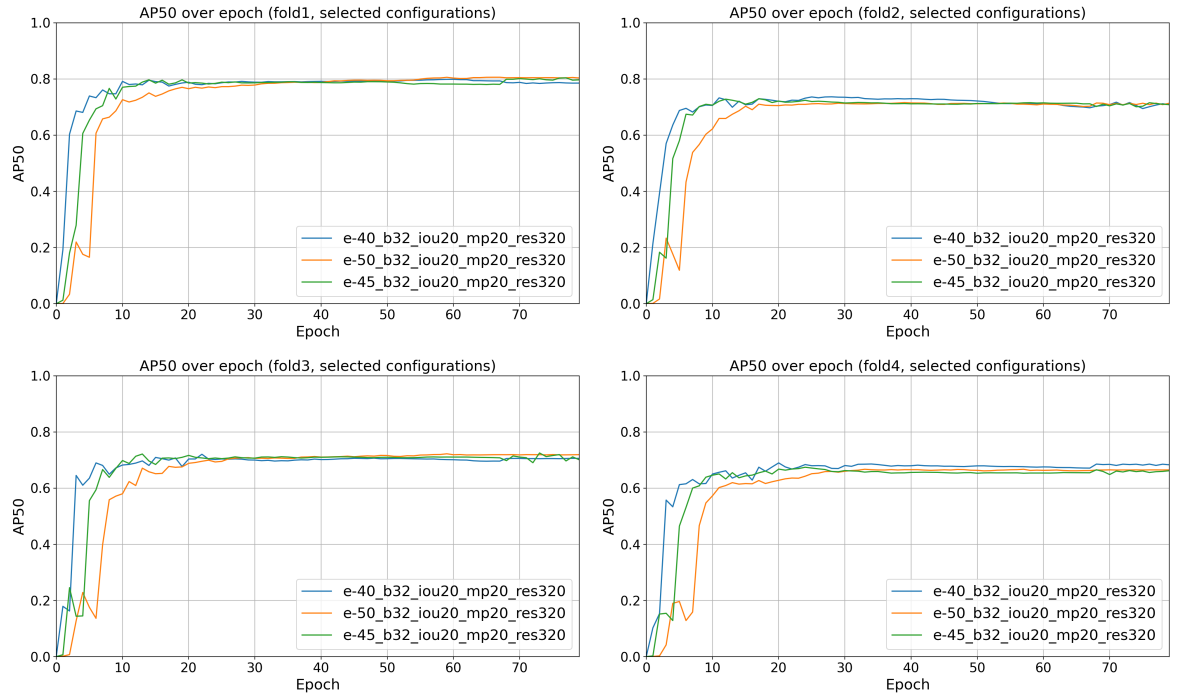


Figure 5.64: Validation AP_{50} over training epochs for the selected learning rates 10^{-4}, $10^{-4.5}$, and 10^{-5} across all folds. The results show faster convergence for 10^{-4}, whereas lower learning rates exhibit delayed recovery while converging towards the same end performance.

The AP_{50} curves show a small dip during the early epochs for $\mathrm{1e{-}4.5}$ and $\mathrm{1e{-}5}$, which aligns with the observed increase in validation loss when augmentations are introduced. Due to slower initial learning, these configurations remain longer in a suboptimal state before recovering.

In contrast, $\mathrm{1e{-}4}$ converges faster and reaches competitive performance earlier. While it exhibits slightly more fluctuation during the initial phase, it stabilizes quickly and surpasses the other configurations around epoch 30. Beyond this point, overfitting becomes more apparent, leading to a gradual decline in AP_{50}.

Given its faster convergence and competitive performance, $\mathrm{1e{-}4}$ was selected for subsequent experiments. The observed overfitting behavior can be addressed in later stages through early stopping and further refinement of augmentation strategies.

Stage 3: Schedule refinement. Based on the flattened AP_{50} curves observed in Figure 5.64, the number of epochs can potentially be reduced. In addition, reducing the number of epochs is expected to limit overfitting while decreasing training time. Therefore, this stage investigates to what extent the training schedule can be shortened.

Given the possibility to run nine experiments in parallel, the total number of epochs was varied as $\{16, 32, 48, 64, 80\}$. In addition, the number of no-augmentation epochs was adjusted to maintain a meaningful proportion of the training schedule at lower epoch counts. Since the previously used value of 12 does not scale well to very short schedules, 8 has also been considered.

For shortest schedule of 16, it should be noted that larger values of $\mathbf{no\_aug\_epoch}$ 8 or 12 represent a substantial fraction of the total training and may dominate the optimization dynamics. Therefore, smaller values 4 and 6 were included to better balance the relative duration of the augmentation phases across different total epoch settings.

The results are shown in Figure 5.65 and Figure 5.66 for no-augmentation epochs of 8 and 12, respectively.

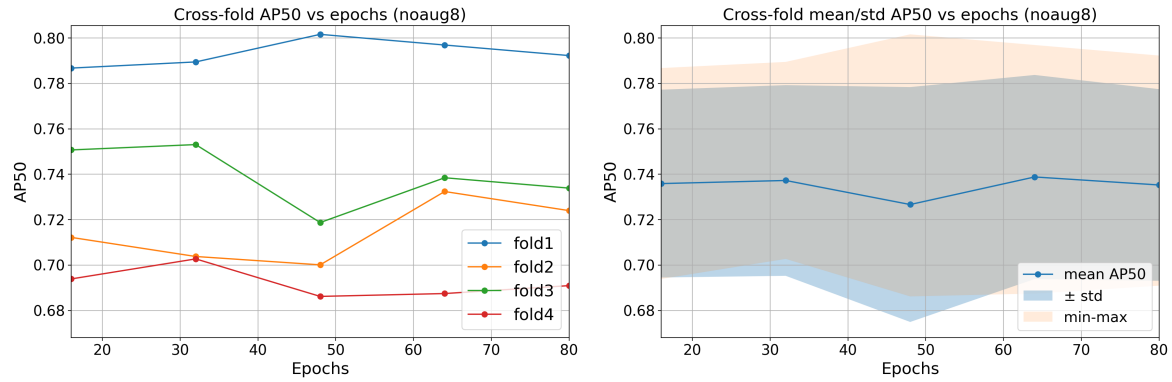


Figure 5.65: Cross-validation AP_{50} as a function of total training epochs with 8 no-augmentation epochs. Left: per-fold performance. Right: mean AP_{50} with variability across folds.

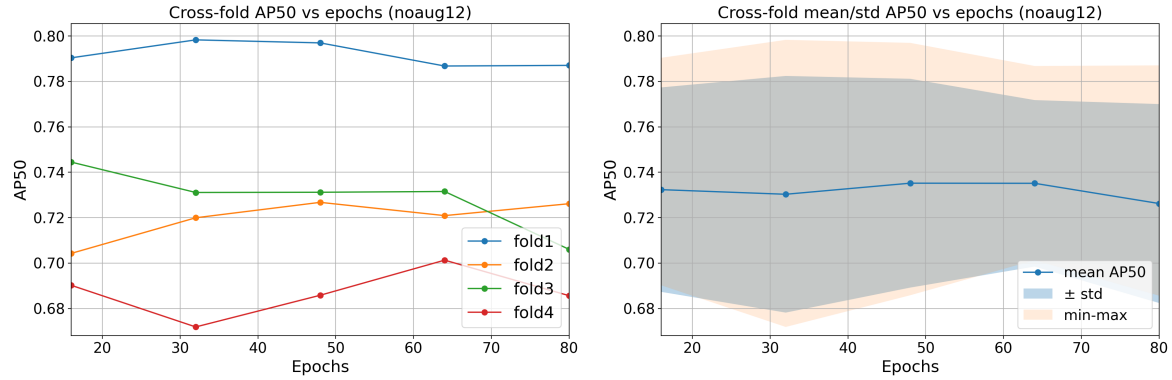


Figure 5.66: Cross-validation AP_{50} as a function of total training epochs with 12 no-augmentation epochs. Left: per-fold performance. Right: mean AP_{50} with variability across folds. It shows plateaus around, 48-64 epochs, indicating no consistent benefit from extending training beyond.

It can be observed that the optimal number of epochs depends on the fold. The relatively easier fold 1 begins to overfit after approximately epoch 50, leading to a reduction in AP_{50}. In contrast, more challenging folds, such as folds 3 and 4, continue to benefit from longer training.This confirms earlier observations that overfitting occurs earlier for easier data distributions, while more difficult folds require longer training to reach comparable performance.

The AP_{50} curves for different epoch settings are shown in Figure 5.67 and Figure 5.68. Across all folds, the overall trends are highly similar, independent of the exact schedule configuration. This indicates that the model reaches its performance plateau early, and that extending training primarily increases overfitting rather than improving generalization. This behavior is most pronounced in folds 3 and 4, where the gap between training and validation performance remains dominant, regardless of the total number of epochs.

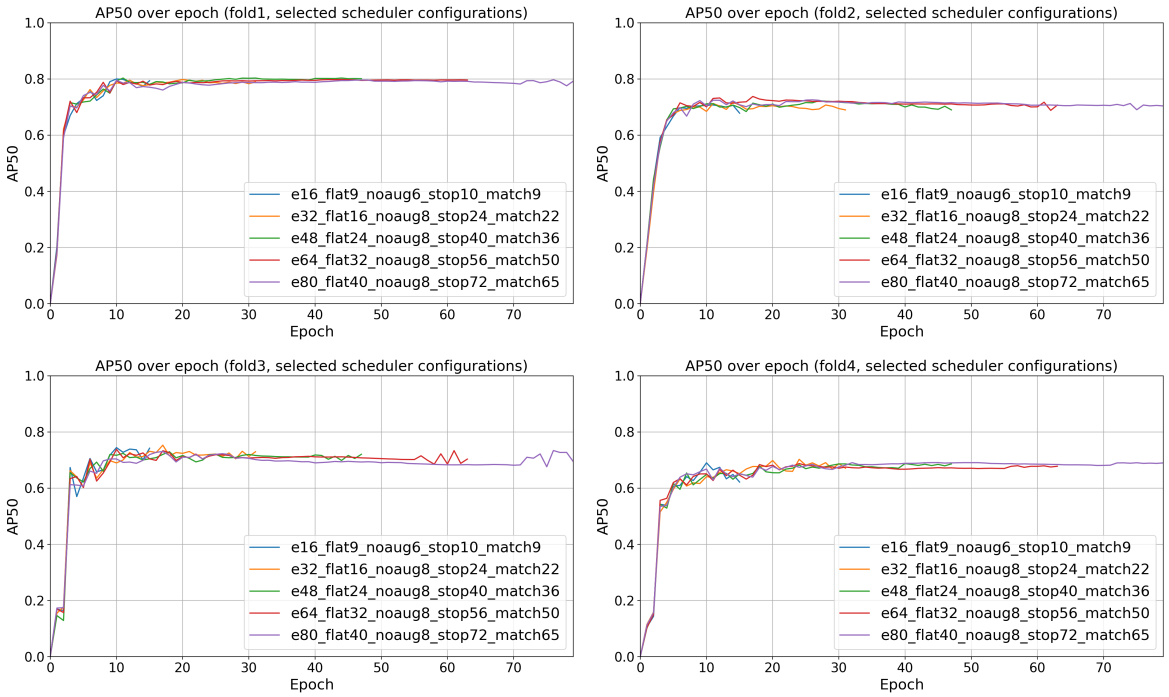


Figure 5.67: Validation AP_{50} over training epochs for selected schedule configurations with 8 no-augmentation epochs across all folds. All configurations converge rapidly within the first 10–15 epochs, with only minor differences thereafter, indicating limited benefit from extended training.

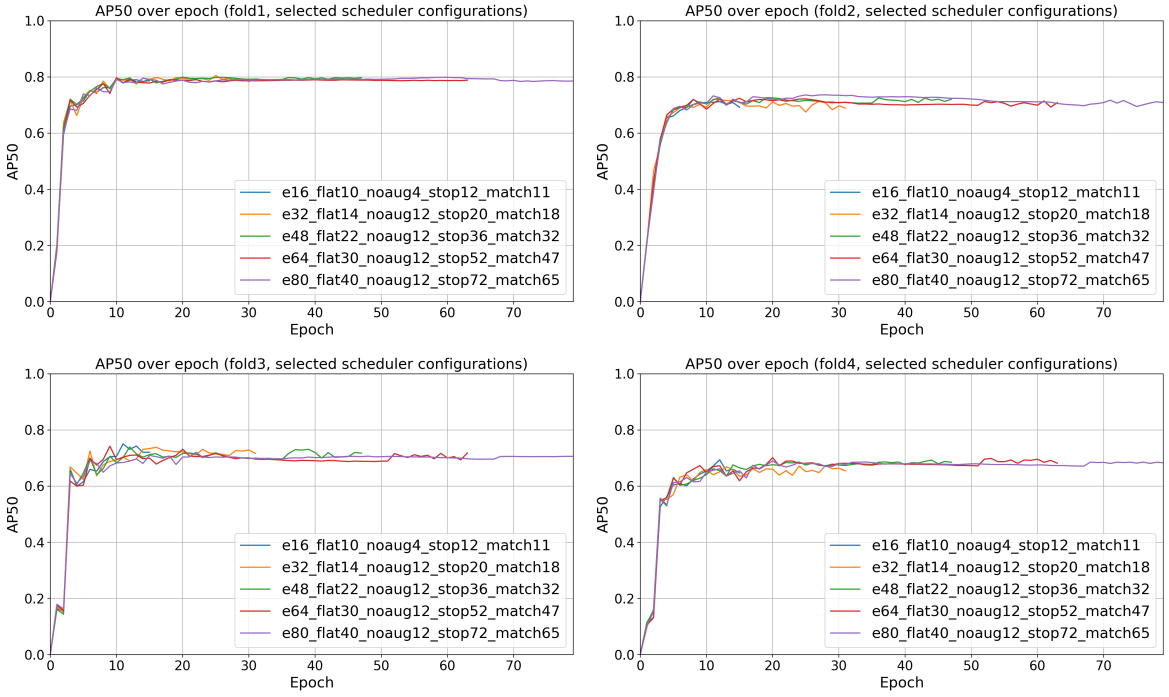


Figure 5.68: Validation AP_{50} over training epochs for selected schedule configurations with 12 no-augmentation epochs across all folds. The convergence behavior closely matches the 8-epoch setting, confirming that extending the no-augmentation phase has minimal impact on overall performance.

Based on Figure 5.65 and Figure 5.66, a configuration with 64 epochs and a no-augmentation phase of 8 epochs achieves the highest mean AP_{50} with relatively low variance across folds. The corresponding loss

curves are shown in Figure 5.69.

With the exception of the initial validation loss increase in folds 3 and 4 and Valdes 2016 generalization gap, the training process remains stable, and convergence is consistently achieved.

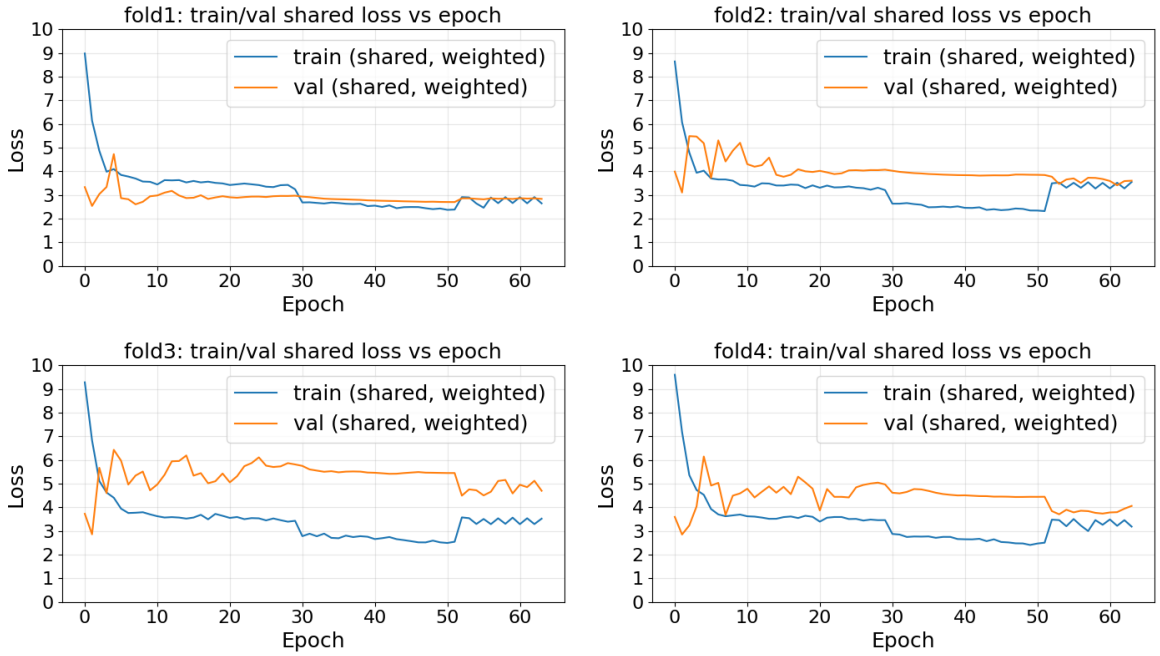


Figure 5.69: Training and validation loss curves for the 64-epoch schedule across all folds. The curves illustrate stable convergence behavior and plateaus after epoch 30-40.

The second-best configuration is obtained with 48 epochs and a no-augmentation phase of 8 epochs, as shown in Figure 5.70. While the training loss remains stable, the validation loss shows increased fluctuations, and signs of overfitting appear earlier, particularly after epoch 20 in folds 2 and 4.

This indicates that even shorter training schedules may be sufficient, especially for easier folds.

Figure 5.70: Training and validation loss curves for the 48-epoch schedule across all folds. It similarly shows plateaus for fold 1 and 3, and overfit for fold 2 and 4 from epoch 25 onwards.

Further reducing the schedule to 32 epochs with 8 no-augmentation epochs gives an AP_{50} close to 0.74, as shown in Figure 5.67, with corresponding loss curves in Figure 5.71. In this configuration, the learning rate decay phase is shortened, and overfitting is reduced.

This suggests that most performance gains are achieved during the early training phase, while later epochs contribute limited improvement and primarily increase overfitting.

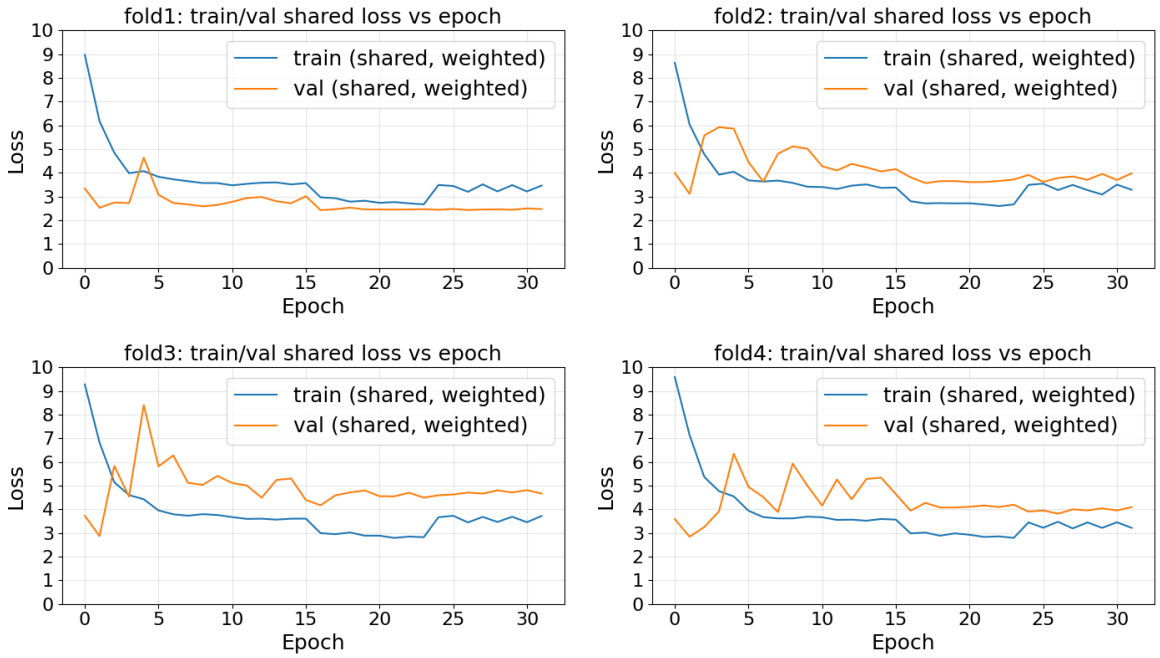


Figure 5.71: Training and validation loss curves for the 32-epoch schedule across all folds. Most of the performance is reached early in training, with the later phases for stabilization and convergence.

While longer schedules achieve slightly higher peak performance, a shorter schedule of 32 epochs provides a favorable trade-off between performance and computational efficiency. This allows for faster experimentation and evaluation of a larger number of hyperparameter configurations.

It should be noted that this schedule is selected for cross-validation experiments only. For the final model, trained on a larger dataset split, fewer epochs are expected to be required due to the increased number of samples per epoch. Therefore, the training schedule will be further adjusted in the final stage.

Stage 4: Augmentation Global Search After fixing the learning rate and schedule settings, the augmentation parameters were refined. This stage was designed as a broad random search over the main DEIM augmentation controls:

$$m \in \{0.0, 0.2, 0.4, 0.8\}, \qquad riou \in \{0.0, 0.2, 0.4, 0.8\},$$

$$mu \in \{0.0, 0.25, 0.5\}, \qquad cb \in \{0.0, 0.25, 0.5\}.$$

For RandomIoUCrop and mosaic probability, the value 0.2 was included because Stage 1 suggested that lower values may be more suitable than the default stronger settings. For MixUp and CopyBlend, the search ranged from the repository default value of 0.5 down to 0.0, in order to test whether reducing or disabling these augmentations improves performance on the present dataset.

Within the available computational budget, 25 random-search configurations were evaluated, in addition to one zero-augmentation baseline. The resulting marginal distribution of the sampled configurations is given in Table 5.21.

Table 5.21: Distribution of Stage 4 random-search runs across augmentation parameter values.

Parameter	Value	Count
Mosaic probability	0	7
	0.2	6
	0.4	6
	0.8	7
RandomIoUCrop probability	0	7
	0.2	6
	0.4	7
	0.8	6
Mixup probability	0	9
	0.25	8
	0.5	9
CopyBlend probability	0	9
	0.25	9
	0.5	8

Looking first at the effect of mosaic probability in Figure 5.72, only small differences in AP_{50} are observed across the tested values. Folds 1 and 2 show a slight preference for lower mosaic probabilities, whereas folds 3 and 4 do not show a consistent trend.

This behavior can be explained by the characteristics of the dataset. The images contain a large proportion of uniform black background at high off-nadir angles, and only a single small object. Mosaic augmentation combines multiple images by tiling them into a single frame, which increases background fragmentation and introduces artificial seams, but does not significantly alter the appearance or frequency of the object itself.

As a result, the overall data distribution remains largely unchanged, and the augmentation neither meaningfully enriches nor degrades the effective learning signal. This explains the marginal differences in performance across the tested values.

Given the absence of consistent improvements, mosaic probability was fixed to 0 for the subsequent refinement stage.

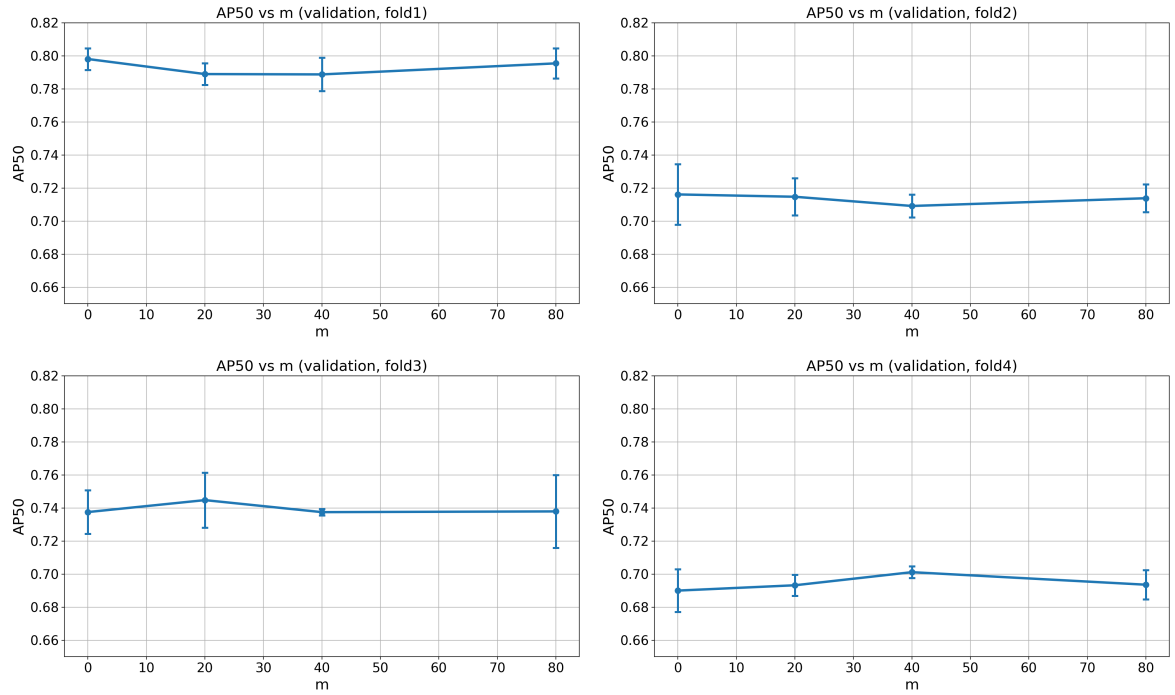


Figure 5.72: Validation AP_{50} as a function of mosaic probability across all folds. Only minor variations are observed, with no consistent trend, indicating that mosaic augmentation has limited impact on performance for this dataset.

The effect of RandomIoUCrop is shown in Figure 5.73. Folds 1 and 2 show a mild preference for values between 0.4 and 0.8, whereas folds 3 and 4 remain comparatively flat and do not exhibit a clear monotonic trend.

In contrast to mosaic, RandomIoUCrop directly alters the spatial focus of the image. By cropping the input, large regions of non-informative background can be removed, increasing the relative size and prominence of the object. This improves the signal-to-noise ratio and can make the object easier to learn.

However, stronger cropping also increases the risk of partially removing the object or placing it near the image boundary, reducing the available feature information. This trade-off explains why intermediate-to-higher cropping probabilities tend to perform better, without showing a strictly monotonic trend.

As a result, although no strong trend is observed, the search is narrowed to intermediate-to-higher values 0.4 and 0.8.

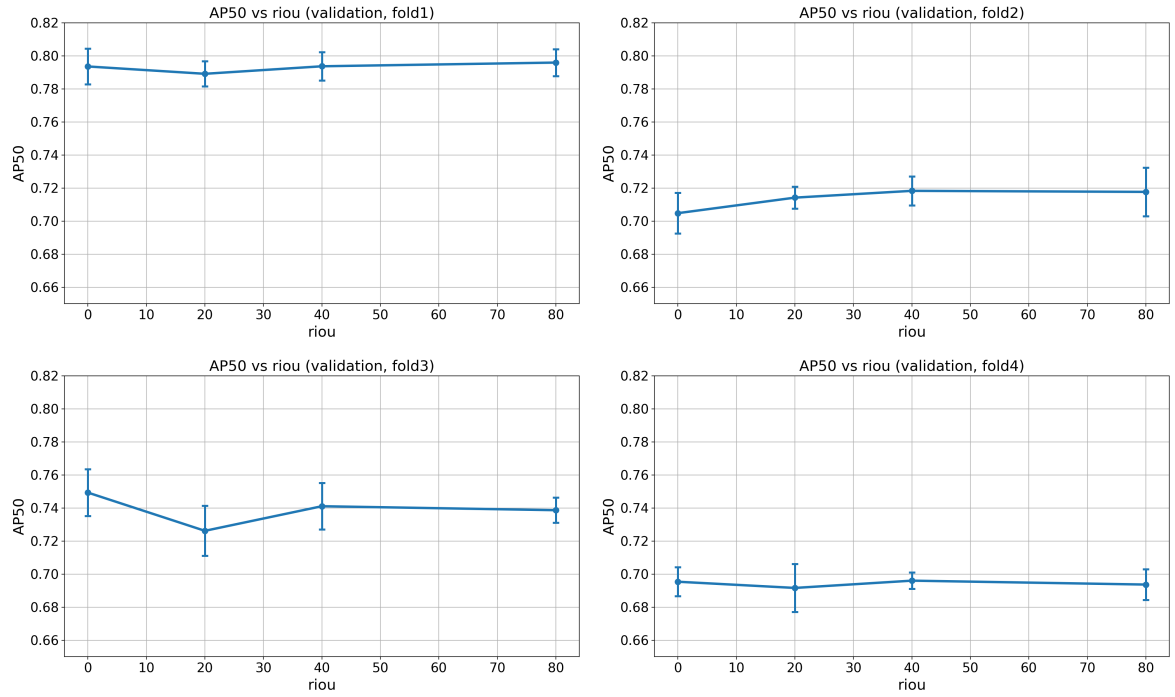


Figure 5.73: Validation AP_{50} as a function of RandomIoUCrop probability across all folds. Intermediate-to-higher values show a slight advantage in some folds, but no consistent monotonic trend is observed.

For MixUp, the fold-wise effects in Figure 5.74 remain small. Fold 1 shows a slight increase towards 0.5, fold 2 also favors higher values, while folds 3 and 4 do not show a consistent improvement.

MixUp blends entire images, preserving global structure but reducing contrast between object and background. In this dataset, where a large fraction of pixels already corresponds to uniform dark background, this blending often does not substantially change the overall image statistics. As a result, the background remains dominant, and the relative object signal is only weakly affected.

At the same time, MixUp acts as a regularizer by smoothing decision boundaries, which can provide small improvements in some folds. This may explain the weak and inconsistent trends across configurations.

Consequently, the search is restricted to 0.25 and 0.5, balancing regularization and signal preservation.

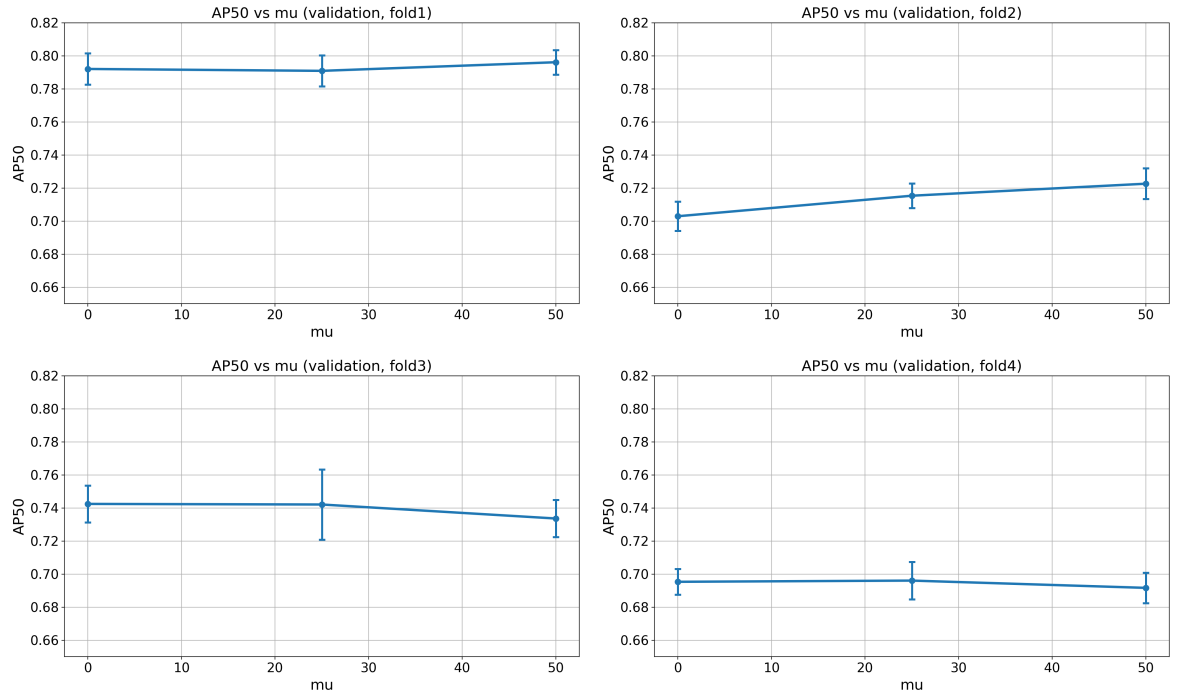


Figure 5.74: Validation AP_{50} as a function of MixUp probability across all folds. The effect remains small, reflecting a weak trade-off between regularization and signal dilution.

The clearest augmentation trend is observed for CopyBlend in Figure 5.75, although the effect remains moderate. Folds 1 and 2 show an increase in AP_{50} towards 0.5, fold 3 is nearly flat with a slight preference for higher values, while fold 4 favors 0.0.

CopyBlend explicitly increases the number of object instances by inserting them into new backgrounds. For a dataset with mostly a single object per image, this effectively enriches the positive sample distribution and exposes the model to more variations in background context. This can improve robustness, especially in simpler folds.

However, the inserted objects may appear in less realistic configurations, particularly against the uniform off-nadir background. This can lead the model to learn simplified or synthetic patterns, which explains the reduced performance at higher values in some folds.

Therefore, the subsequent search is restricted to CopyBlend probabilities of 0.25 and 0.5.

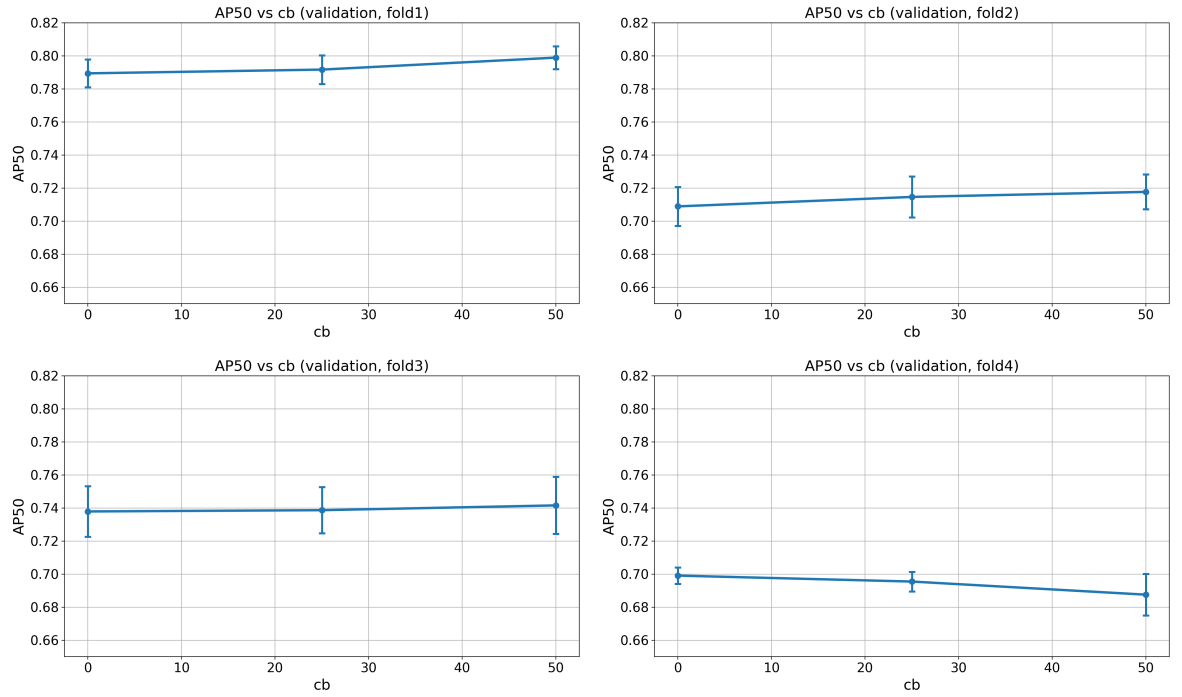


Figure 5.75: Validation AP_{50} as a function of CopyBlend probability across all folds. Moderate improvements are observed in fold 1, 2, and 3 folds at higher values, although the effect remains limited and fold-dependent.

Stage 5: Augmentation Local Search Combining the results from the previous stages, a local refinement search was performed over the narrowed augmentation ranges. The searched parameter ranges were

$$m = 0, \qquad riou \in \{0.4, 0.8\}, \qquad mu \in \{0.25, 0.5\}, \qquad cb \in \{0.25, 0.5\}.$$

The resulting mean validation metrics across folds are given in Table 5.22, with the full results available in Appendix E.

Table 5.22: Mean Stage 5 validation results across folds.

ID	Config settings	AP_{50}	$AP_{50:95}$	AP75	Best F1	F1 thr	Latency [ms/img]
02	m=0.0, riou=0.8, mu=0.5, cb=0.5	0.732	0.324	0.235	0.936	0.341	16.199
03	m=0.0, riou=0.4, mu=0.25, cb=0.25	0.725	0.320	0.226	0.933	0.377	44.387
04	m=0.0, riou=0.8, mu=0.25, cb=0.25	0.724	0.320	0.231	0.937	0.347	43.571
05	m=0.0, riou=0.4, mu=0.5, cb=0.5	0.736	0.328	0.234	0.939	0.329	16.597
06	m=0.0, riou=0.8, mu=0.5, cb=0.25	0.732	0.322	0.227	0.936	0.340	21.678
07	m=0.0, riou=0.4, mu=0.25, cb=0.5	0.733	0.324	0.235	0.938	0.339	15.788
08	m=0.0, riou=0.4, mu=0.5, cb=0.25	0.731	0.323	0.226	0.935	0.356	41.062
09	m=0.0, riou=0.8, mu=0.25, cb=0.5	0.737	0.328	0.242	0.936	0.324	27.854

The local augmentation search produced only small differences in validation performance. Across the eight evaluated configurations, mean AP_{50} varies only between 0.724 and 0.737, while $AP_{50:95}$ and AP_{75} also remain within narrow ranges. This indicates that, within the narrowed search space, augmentation settings act only as a weak lever on final detector performance.

This observation limits how strongly individual configurations can be ranked. The difference between the best and worst AP_{50} values is comparable to the level of variation typically observed between repeated runs of similar configurations. Therefore, the Stage 5 results should be interpreted mainly as a confirmation that performance is relatively insensitive to moderate augmentation changes, rather than as evidence for a sharply defined optimum.

This limited sensitivity is further illustrated by configurations that were repeated between Stage 4 and Stage 5, as summarized in Table 5.23. For these repeated settings, the difference in AP_{50} between two runs of the same configuration can be as large as 0.014, which is comparable to or larger than the difference between distinct augmentation settings in Stage 5 itself.

Table 5.23: Comparison of repeated augmentation configurations between Stage 4 and Stage 5.

Same configuration	Stage 4	Stage 5	Mean AP_{50} change	Mean $AP_{50:95}$ change	Mean AP75 change	Mean latency change [ms/img]
m=0.0, riou=0.8, mu=0.5, cb=0.5	ID 02	ID 02	-0.004	-0.001	+0.002	-25.330
m=0.0, riou=0.4, mu=0.5, cb=0.5	ID 11	ID 05	-0.011	-0.006	-0.007	-18.136
m=0.0, riou=0.8, mu=0.5, cb=0.25	ID 06	ID 06	-0.014	-0.002	-0.003	-28.855
m=0.0, riou=0.4, mu=0.25, cb=0.5	ID 17	ID 07	-0.009	-0.003	+0.005	-25.366

This run-to-run variation is expected in stochastic deep learning pipelines. Even when the hyperparameters are identical, random initialization, stochastic mini-batch ordering, random augmentation sampling, and non-deterministic GPU execution can lead to slightly different optimization trajectories and therefore to different final validation metrics [166].

A similar caution applies to the reported latency values. The latency per image depends not only on the network architecture and input size, but also on system-level effects like GPU load and memory allocation state usage during evaluation. As a result, large latency differences between otherwise similar runs should not be interpreted as being caused solely by augmentation settings.

Taken together, these results suggest that augmentation choices are only a secondary factor in the present setting. One likely reason is that the strongest augmentations are active only during a limited portion of the training schedule. In addition, the dataset already contains substantial variability in whale object size, orientation, and radiometric conditions due to different input resolutions and the off-nadir rendering pipeline itself. This reduces the relative need for strong synthetic augmentation.

The low spatial resolution of the input images further limits the extent to which aggressive augmentation can introduce meaningful new structure. Specifically, when the object occupies only a small region of the image, stronger transformations can alter the background more than the object signal. Moreover, the task is single-class detection, so the model mainly learns to separate whale-like patterns from background without having to distinguish between multiple visually similar classes. This reduces the dependence on strong augmentation diversity compared to more complex multi-class detection settings.

At the same time, the results do not support removing augmentation altogether. Although the average differences are small, the zero-augmentation baseline does not emerge as a clearly superior setting, and in some folds it performs worse than the moderate augmented configurations. This indicates that some augmentation remains beneficial, even if its effect is modest.

Given the weak sensitivity of performance to the exact augmentation values, the final selection should not be interpreted as a decisive optimum. Among the Stage 5 configurations, ID 09 achieves the highest AP_{50} and also the highest AP_{75}, while remaining close to the other top-performing runs in $AP_{50:95}$ and F1-score. Although its advantage over configuration ID 05 is only 0.001 in AP_{50} and is therefore not statistically meaningful on its own, it is consistent with a slight improvement in localization quality as reflected by AP_{75}.

For that reason, configuration ID 09 was selected as the final augmentation setting. This corresponds to

$$m = 0.0, \qquad riou = 0.8, \qquad mu = 0.25, \qquad cb = 0.5.$$

This selection should be interpreted as a reasonable final operating point within a broad plateau of similar-performing augmentation settings. In practice, the exact choice of augmentation parameters within this range has only a minor impact on the final model performance.

5.4.4. Final Model

After selecting the final hyperparameter configuration, the best-performing setting was deployed for final model training.

For operational deployment, maintaining multiple cross-validation models is impractical due to onboard computational constraints. Therefore, a single model was trained on nearly the full dataset, using approximately 95% of the available data for training and 5% for validation and early stopping. This maximizes the amount of training data while retaining a minimal mechanism to monitor convergence.

However, increasing the training set size changes the number of samples processed per epoch. Consequently, keeping the same number of epochs as in cross-validation would increase the total number of optimizer updates and introduce a higher risk of overfitting. To account for the increased training set size in the final configuration, the effective number of training samples per epoch was quantified.

Across the seven non-test locations, the dataset contains 11,633 images. During cross-validation, approximately two out of seven locations were held out for validation in each fold, resulting in an effective training size of

$$N_{\text{train,cv}} \approx \frac{5}{7} \cdot 11633 = 8309. \tag{5.93}$$

For the final training setup, 95% of the dataset was used, yielding

$$N_{\text{train,final}} = 0.95 \cdot 11633 \approx 11051. \tag{5.94}$$

This corresponds to an increase in training samples per epoch by a factor of

$$\frac{N_{\text{train,final}}}{N_{\text{train,cv}}} = \frac{11051}{8309} \approx 1.33. \tag{5.95}$$

Assuming approximately constant gradient statistics per batch, maintaining a comparable number of optimizer updates, the number of epochs was reduced accordingly:

$$E_{\text{final}} = \frac{32}{1.33} \approx 24. \tag{5.96}$$

Therefore, a final model was trained on 24 epochs on the 95% split, with the remaining 5% used for validation. Table 5.24 summarizes the resulting architecture and training configuration of the final model.

Table 5.24: Final training configuration after hyperparameter optimization. Only parameters modified relative to the default DEIMv2 configuration are listed.

Setting	**Final value**
Input resolution	256×256
Batch size	32
Base learning rate	1×10^{-4}
Backbone learning rate	1×10^{-5}
RandomIoUCrop probability	0.8
Mosaic probability	0.0
Mixup probability	0.25
CopyBlend probability	0.5
Normalization	Custom to dataset
Total epochs	24
Flat epoch	12
No-augmentation epochs	8
Augmentation policy epochs	[4, 12, 16]
Mixup epochs	[4, 12]
CopyBlend epochs	[4, 16]
Stop epoch (strong augmentation)	16
Matcher change epoch	14
IoU order α	4.0
EMA decay	0.9999

The resulting AP_{50} and training and validation loss curves are shown in Figure 5.76.

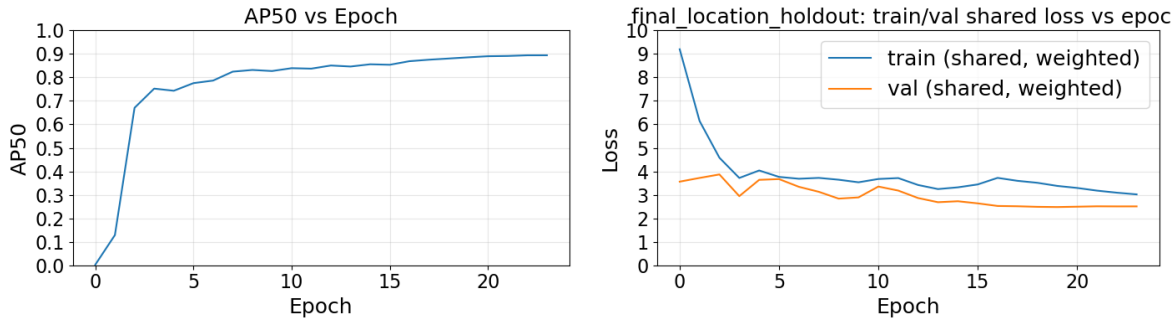


Figure 5.76: Training and validation AP_{50} together with the shared loss over epochs for the final 24-epoch model. Both metrics plateau without divergence, indicating convergence and an appropriate stopping point.

The training and validation loss curves in Figure 5.76 exhibit increases at epochs 4, 12, and the former also at 16, corresponding to changes in augmentation strength. Following these transitions, the validation loss decreases steadily and reaches a plateau around epoch 18, indicating that convergence is effectively achieved. Throughout training, the validation loss remains consistently below the training loss, which can be attributed to the stronger data augmentation applied during training, increasing the difficulty of the training samples relative to the validation set.

The AP_{50} curves show rapid improvement during the initial epochs, followed by gradual saturation. Beyond approximately epoch 19, performance gains become marginal, indicating diminishing returns from further training. Although convergence is effectively reached at this point, a total of 24 epochs was retained to ensure stable performance. The resulting metrics are summarized in Table 5.25.

Table 5.25: Final model performance after 24 epochs.

ID	Set	AP_{50}	$AP_{50:95}$	AP75	F1	Thr
	VAL	0.894	0.448	0.443	0.962	0.380

The decomposition of the loss function in Figure 5.77 shows that the matchability-aware loss (MAL) dominates throughout training. This indicates that optimization is primarily driven by classification performance, while the contributions from bounding box regression losses (L1 and GIoU) remain comparatively small. This behavior is consistent with the emphasis on AP_{50}, where correct detection is more critical than precise localization.

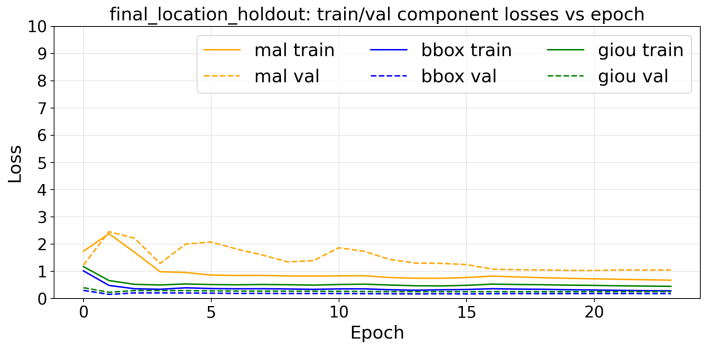


Figure 5.77: Decomposition of training and validation loss components over epochs for the final model. The matchability-aware loss (MAL) is the dominant contributor, whereas bounding box (L1) and GIoU losses remain low, indicating that optimization is primarily driven by classification-related errors.

Overall, the final model demonstrates stable convergence, strong performance on the validation set, and no evidence of detrimental overfitting. Although convergence is effectively reached around epoch 19, the

remaining epochs provide small but consistent improvements in AP_{50}. While it may be possible to reduce the number of epochs without significantly degrading performance, determining an optimal stopping point would require additional tuning and introduce sensitivity to this choice. Given that the expected gains are marginal, this fine-tuning is not justified within the scope of this work. Therefore, the full 24 epochs are retained. Final performance will be assessed on the held-out test set.

Model Size Comparison

After selecting the final hyperparameter configuration, an additional comparison was made across different DEIM model sizes: Atto, Femto, Pico, N, S, M, L, and X, to examine the trade-off between detection performance and inference latency.

An overview of the approximate model characteristics reported by the repository is given in Table 5.26. The $AP_{50:95}$ values were used as a reference to motivate the comparison, while the actual whale-detection performance was evaluated using the custom dataset and selected baseline training recipe, as a function of parameter count and GFLOPs.

Table 5.26: Overview of DEIM model sizes and their reported characteristics on COCO, adapted from the DEIM model zoo.

Model	AP (COCO)	#Params	GFLOPs	Latency (ms)
N	43.0	3.6M	6.8	2.32
S	50.9	9.7M	25.6	5.78
M	53.0	18.1M	52.2	8.80
L	56.0	32.2M	96.7	10.47
X	57.8	50.3M	151.6	13.75

Using the final baseline hyperparameter configuration, each of these model sizes was trained and evaluated to determine how the whale-detection performance scales with increasing model capacity. The comparison focused on the validation AP_{50} to assess whether larger models provide a meaningful increase in detection performance for this specific application. Latency could not be evaluated reliably, as the available measurements during training were affected by varying GPU utilization and therefore not representative of inference conditions. The complete set of results is provided in section E.3.

The resulting AP_{50} and $AP_{50:95}$ scores for the range of model sizes is given in Figure 5.78. In general, increasing model size is expected to improve representational capacity and thus detection performance, but also to increase computational cost and inference time. This trade-off is particularly relevant for the AI-based Tip and Cue concept, where onboard processing must remain sufficiently lightweight to support timely tasking decisions.

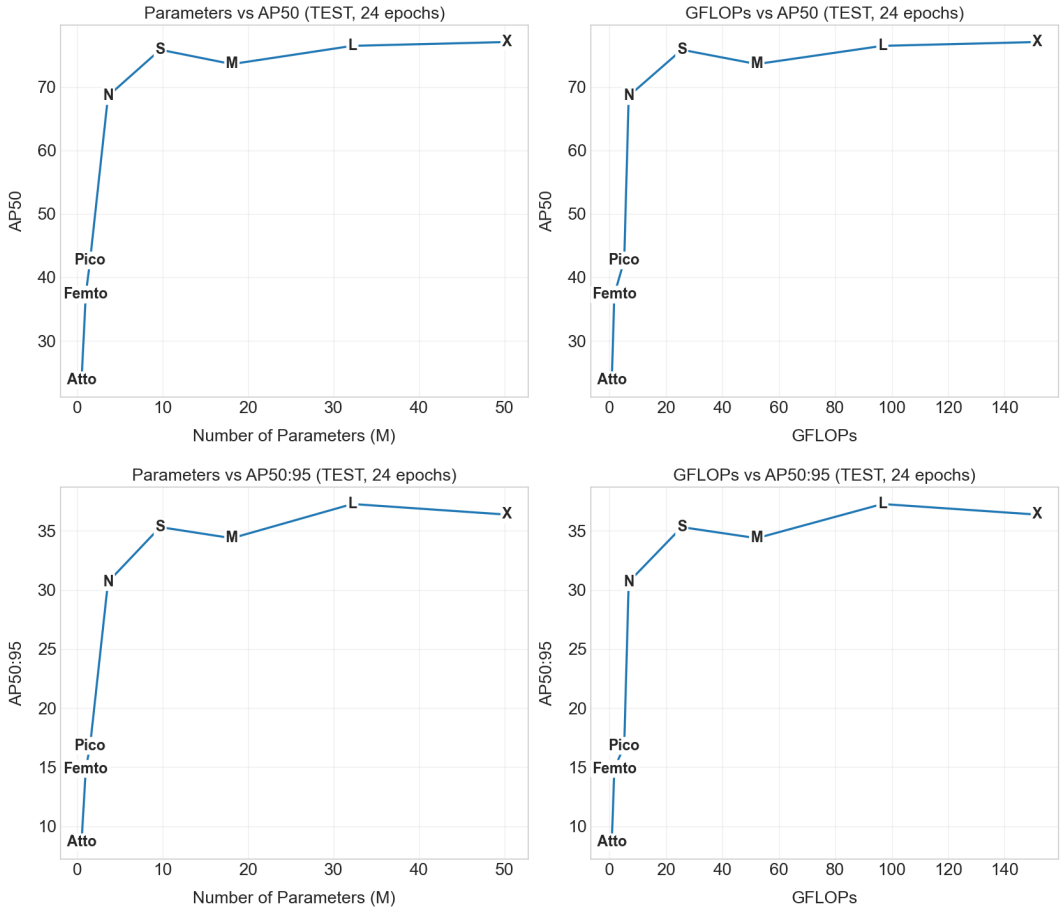


Figure 5.78: Scaling behaviour of DEIM model sizes on the whale detection task. AP_{50} (left) and $AP_{50:95}$ (right) are shown as a function of parameter count and GFLOPs. In general, detection performance improves with increasing model size, where the S model performs particularly well given that that the hyperparameter optimization was carried out for this model and subsequently transferred to the other scales.

It should be noted that the hyperparameter optimization described in subsection 5.4.3 was carried out for the S model only and subsequently transferred to the other model scales. As a result, the comparison shall only be used as a relative estimate between the models, but not for final model selection.

The results show an overall trend of increasing performance with model size. Excluding the S model, both AP_{50} and $AP_{50:95}$ generally increase from the smallest variants toward the larger models. The S model performs particularly well relative to its size, outperforming the larger M model and approaching the performance of L, which is consistent with the fact that the hyperparameter optimization was conducted specifically for this model and subsequently transferred to the other scales. Similarly, L outperforms X, showing that local deviations from the overall scaling trend remain present. Overall, the results suggest that model scaling improves detection performance in general, while the exact ranking between neighbouring model sizes is also influenced by the transferred training configuration.

A dedicated re-optimization of hyperparameters and training schedules for each model size could potentially improve their absolute performance. However, given that the primary objective of this analysis is to assess relative scaling behaviour under a consistent setup, additional tuning is not pursued.

5.5. Experimental Setup

After training the final model, the mission simulations can be performed. This section describes the experiments conducted to evaluate imaging effects, mission geometry, and the performance of AI-Based Tip and Cue relative to alternative satellite architectures. In line with the methodological framework in subsection 5.1.1, the experiments are structured so that detector-level effects and mission-level effects

can be assessed separately, and then combined through the benchmark.

5.5.1. Imaging Effects

To evaluate the effect of Cue off-nadir imaging on onboard detection performance, two imaging experiments are defined. First, it is evaluated to what extent representative modelling of both geometric and radiometric off-nadir effects contributes to actual onboard detection performance. In other words, this experiment examines the gap between the standard training approach in literature, where images are typically treated as nadir and radiometrically corrected, and the representative off-nadir imaging conditions expected in an actual Tip-and-Cue mission. Second, after including the geometric effects, it is evaluated how important it is to include radiometric effects in the modelling as well.

To do so, two additional models are trained on top of the final representative model from subsection 5.4.4. To evaluate the omission of both geometric and radiometric effects, one model is trained on the raw Whales in Space dataset, where only the crop, rotation, and mirror operations are applied to generate comparable training patches. In parallel, to evaluate the omission of radiometric modelling only, a second model is trained on the off-nadir translated raw patches, thus including the geometric distortions but excluding the radiometric diversity introduced by the rendering pipeline. In this way, the contribution of geometric and radiometric modelling can be separated.

Both additional models are trained using the same training recipe as the representative model, on 95% training data with 5% validation data for early stopping. With that, they go through the same processing chain as the representative model trained in subsection 5.4.4, ensuring that the comparison is not affected by differences in optimization procedure or dataset size.

After training, three models are available: the full representative model, one without radiometric effects, and one without both radiometric and geometric effects. Their performance is then evaluated on the same held-out test datasets, consisting of Auckland2006 and Pelagos2016, representing locations unseen during training, similar to actual operational scenarios. Because the first experiment isolates detector performance, the models are compared primarily using AP_{50} on the test dataset.

After that, it is evaluated how training on data without representative radiometric diversity affects performance when the detector is deployed in an operational mission scenario. This reflects a typical design pipeline in which a model is developed on simplified training data and then applied under actual off-nadir imaging conditions. To do so, the trained models are deployed on the mission-specific dataset generated for the baseline mission configuration. Following the benchmark defined in subsection 5.1.1, performance in this imaging-effects comparison is evaluated using the successful detection capacity C_{succ}, complemented by the detector-level metrics AP_{50}, precision, and recall. Because the orbital configuration is kept fixed and only the onboard detection model is varied, the response metrics latency L and viewing time V are identical across cases and are therefore not used to distinguish performance in this experiment.

5.5.2. Mission Geometry

Next, mission simulations are performed to evaluate how the orbital geometry affects end-to-end Tip-and-Cue performance. More specifically, by varying one parameter at a time with respect to the baseline configuration, the effect of the maximum allowed off-nadir angle and the time delay between Tip and Cue can be evaluated in terms of latency, viewing time, and final detection performance.

For the maximum off-nadir angle experiment, the requirement is varied from 5 to 60 degrees with a step size of 5 degrees, while keeping the Tip-Cue time delay fixed at 5 minutes. For the time-delay experiment, the delay between Tip and Cue is varied from 1 to 10 minutes with a step size of 1 minute, while keeping the maximum allowed off-nadir angle fixed to the baseline requirement of 40 degrees. In this way, the effect of look-angle flexibility and the effect of orbital timing can be studied separately. These experiments are performed separately for the two source-resolution groups introduced in subsection 5.4.1. In addition, the detection evaluation is carried out only on the held-out test data locations, to ensure the reported performance reflects generalization to unseen locations.

Each simulation is repeated three times with seeds 1, 17, and 42, in order to reduce the effect of variance caused by stochastic whale initialization and propagation. All benchmark results are therefore reported based on repeated runs, for which the average is taken.

Each mission generates the outputs described in subsection 5.1.3. From the resulting mission overview file, the event-level benchmark metrics can be obtained, including the mean latency L and mean viewing time V as defined in subsection 5.1.1. In addition, the geometric detection parameters of each Cue acquisition are logged. Based on these logged parameters, a mission-specific detection dataset is generated for the corresponding test group, after which the final representative model is evaluated on that dataset using the fixed deployment threshold selected during model tuning.

With that, the full end-to-end benchmark can be computed per configuration. In particular, the successful detection capacity C_{succ} is derived at mission level, while L and V describe the response characteristics of the successful detections. Detector-level performance is reported separately using AP_{50}, precision, and recall. Plotting these quantities across the off-nadir-angle and time-delay sweeps makes it possible to identify the trade-off between faster response, longer viewing opportunity, and degradation in detection performance at more demanding viewing geometries.

5.5.3. Comparative Evaluation

Besides varying parameters within one Tip-and-Cue configuration, it is also aimed to compare AI-Based Tip and Cue missions with respect to alternative satellite architectures. To do so, two Walker-Delta reference constellations of high-resolution Cue satellites are simulated, namely a 8×2 and a 4×4 configuration. Both are initialized from the same Cue orbital conditions around which the baseline mission is built, representing WorldView-3-like orbital geometry.

In addition, an independently operating Tip-and-Cue configuration is simulated, with six Tip satellites based on Sentinel-2A, Sentinel-2B, Sentinel-2C, and Landsat-7, Landsat-8, and Landsat-9, and six Cue satellites based on WorldView Legion 1-6. Their orbital parameters are derived from these real systems, while the benchmark swath width and spatial resolution are retained to isolate the effect of mission geometry. In this way, the independently operating case represents a realistic non-coordinated architecture against which the coordinated Tip-and-Cue configurations can be compared.

Lastly, to examine the full system potential beyond a single Tip-and-Cue pair, a Tip-and-Cue constellation is simulated as well, with four orbital planes and two Tip satellites and two Cue satellites per plane. This configuration also totals 16 satellites and therefore enables direct comparison with the 16-satellite VHR constellation. In this way, it can be evaluated how much performance changes if half of the VHR constellation capacity is replaced by wide-swath Tip satellites that improve search capability and tasking efficiency.

The same steps as in subsection 5.5.2 are then repeated for each configuration. First, the orbital simulation produces the observation opportunities, latency statistics, and acquisition geometries. Second, the corresponding mission-specific detection datasets are generated from the test locations. Third, the appropriate detection model is deployed on those datasets and the benchmark metrics are computed. For the AI-Based Tip and Cue configurations, the final representative detection model is used, since these missions rely on off-nadir Cue acquisitions. For the VHR reference constellations, a separate detection model trained on nadir-only reflectance data with radiometric effects is used instead. In this way, each architecture is evaluated with a detector matched to its imaging conditions, enabling equal comparison than deploying the same off-nadir-trained model on both cases.

Comparing C_{succ}, L, V, AP_{50}, precision, and recall across configurations makes it possible to assess which architecture performs better in absolute terms and the response and detection performance associated with each configuration. Specifically, comparing the 16-satellite VHR Walker constellation with a single Tip-and-Cue configuration based on the successful detection capacity metric enables an estimate of how many satellites can be saved.

5.5.4. Automation Pipeline

To ensure efficient execution of the full set of experiments, an automation pipeline is defined to run all simulations and detection evaluations automatically. This pipeline is organized around three folders: `waiting`, `running`, and `done`. In the `waiting` folder, the `settings.py` file is placed that specifies one simulation run, including the delay between Tip and Cue, the maximum off-nadir requirement, the number of satellites, the number of orbital planes, and other mission parameters.

The generation of these settings files is automated as well. In particular, scripts are generated automatically

for the parameter sweeps over off-nadir angle, from 5 to 60 degrees in steps of 5, and over latency, from 1 to 10 minutes in steps of 1. A naming convention is used to identify each run. For example, `TC_4x2sat_40deg_5min_1sd` represents a Tip-and-Cue simulation with 4 orbital planes, 2 satellites per plane, a maximum off-nadir requirement of 40 degrees, a 5-minute Tip-Cue delay, and whale initialization seed 1.

Running the master script sequentially starts subprocesses that copy the selected settings file to the main AI-Based Tip and Cue folder and execute the corresponding `run_simulation` script. During execution, the settings file is moved from `waiting` to `running` and finally to `done`, so that the simulation status can be tracked directly from the folder structure.

For each run, results are generated automatically in the results folder. Each executed simulation produces a 2D footprint map of the Tip and Cue coverage, simulation logs, plots of off-nadir angle, latency, and viewing-time distributions, a results overview, a movie visualizing the full simulation, and, optionally, the off-nadir imagery at the detection instances as discussed in subsection 5.1.3. If the off-nadir imagery has not yet been generated during the simulation itself, it can also be created afterwards from the logged observation geometry.

Lastly, a separate script loads the generated imagery together with the selected model into the detection pipeline. After applying the fixed IoU threshold and operating confidence threshold, the model is evaluated on the mission-specific dataset and the resulting predictions are written back to the benchmark outputs. In this way, the orbital simulation, image generation, and detection evaluation are executed as one reproducible experiment chain.

5.5.5. Off-Nadir Sample Availability

At large off-nadir angles, whale instances can become too small, causing the bounding box corner points to overlap. In these cases, the image generator fails, and these samples are skipped, as described in subsection 5.4.1. As a result, within the dataset at high off-nadir angles, only the remaining feasible (easier) samples are included, while the difficult cases are implicitly removed. This biases the detector evaluation if not corrected.

To account for this, the number of successfully generated whale samples per location and off-nadir angle is reported in Table 5.27. For small off-nadir angles, the number of samples remains approximately constant. At larger angles, the number decreases, with a clear drop beyond $50°$ and earlier degradation for lower-resolution sources (WV2, GE1, QB2).

Table 5.27: Number of successfully generated whale samples per location and off-nadir angle.

Location	5°	10°	15°	20°	25°	30°	35°	40°	45°	50°	55°	60°	Total samples
Auckland2006	69	69	69	69	69	69	69	69	69	66	60	43	69
Auckland2011	32	32	32	33	31	33	31	32	32	32	27	14	34
Ignacio2017	79	79	79	80	78	79	79	79	79	79	73	60	80
Maui2015	56	56	56	56	56	56	56	56	55	52	51	40	56
Pelagos2016	32	32	32	32	32	32	32	32	32	31	29	26	34
Valdes2012	81	81	81	81	81	81	81	80	80	80	70	34	84
Valdes2014	58	58	58	59	57	59	57	58	58	57	57	49	59
Valdes2016	120	120	119	122	119	121	119	119	119	117	111	90	129
Witsand2009	88	88	88	89	87	88	88	88	85	82	71	33	88
Total	615	615	613	623	609	620	610	613	609	596	549	389	633

For each detection within the off-nadir experiments, the fraction of successfully generated samples is computed as the ratio between the generated samples and the total number of samples per location. For example, for Auckland2006 at $60°$, this gives $43/69 = 0.623$. These fractions were verified by inspecting the dataset, where missing samples correspond to cases where the whale becomes too small and disappears at lower off-nadir angles.

Since the detector is evaluated only on successfully generated samples, a correction is applied during evaluation. Each whale instance is retained with probability equal to the corresponding success fraction. If

the sample is removed, it is treated as a missed detection. This ensures that unavailable whale instances are accounted for as false negatives.

False positives are not affected. Ocean samples are not corrected, since their generation failures are caused by oversized whales rather than disappearance, and therefore do not bias detection performance.

6

Verification and Validation

This chapter verifies and validates the three components of the AI-Based Tip and Cue framework and then demonstrates their integration. First, section 6.1 verifies the orbital simulation framework through controlled geometric cases, analytical checks, and mission-level visualization. Next, section 6.2 validates the off-nadir imaging pipeline through internal consistency checks and comparison with Φ-Sat-2 data. Then, section 6.3 evaluates the onboard detection pipeline, including split integrity, detector performance, and cross-validation behaviour. Finally, section 6.4 demonstrates the end-to-end pipeline in one baseline AI-Based Tip and Cue scenario.

6.1. Orbital Simulation

This section verifies whether the orbital simulation framework behaves consistently with the intended AI-Based Tip and Cue mission logic. The objective is to test whether the implemented geometry, tasking, pointing, and logging produce internally consistent outputs under controlled cases. To do so, the framework is verified stepwise, from initialization and orbital propagation to field-of-view projection, tasking, pointing, visualization, and data logging.

6.1.1. Simulation Setup

Initial Position

To validate the initialization of the satellite positions, a satellite was requested at latitude and longitude $(0.0,\ 0.0)$ with an altitude of $616\,\mathrm{km}$, while the target was placed at $(0.0,\ 0.0,\ 0.0\,\mathrm{m})$. The simulation time was set to 12:00 on 20 March 2025, which lies close to the vernal equinox occurring at 09:01. At this time, the satellite location should be fully illuminated and located directly above the intersection of the Greenwich meridian and the Equator.

The resulting visualization is shown in Figure 6.1. From the figure it can be observed that both the satellite and the target are positioned exactly at Greenwich and the Equator. Furthermore, the solar illumination direction aligns with the expected equinox geometry, where the sun is perpendicular to the prime meridian. This confirms that the coordinate initialization and the associated reference frame transformations correctly place the satellite and target within the simulation environment.

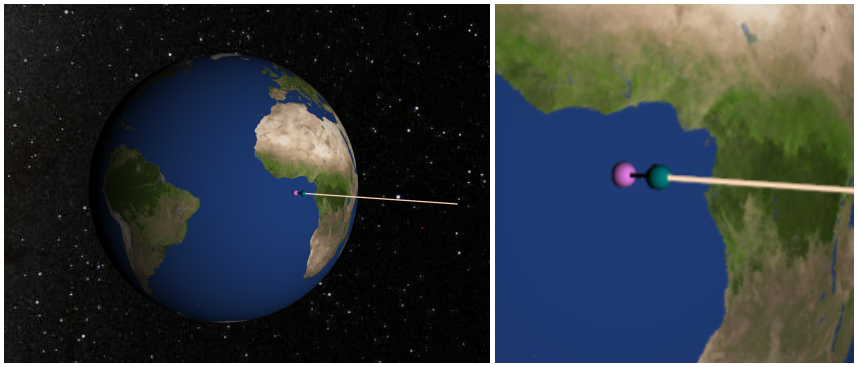


Figure 6.1: Satellite and target at zero latitude and longitude, near the vernal equinox on 20 March 2025. It can be observed that both are placed exactly at Greenwich and the Equator, while the sun direction aligns with the expected illumination geometry.

A second verification was performed under an arbitrary geographical configuration. In this case, the target position was set to Rome, Italy at $(41.902782,\ 12.496366)$ latitude and longitude, while the satellite initial position was defined above Amsterdam at $(52.377956,\ 4.897070)$ latitude and longitude with an altitude of $617\,\mathrm{km}$ [167]. The simulation time was chosen as 21 June 2025, corresponding to the summer solstice period, where nautical twilight occurs at approximately 03:36 AM [168].

The resulting configuration is illustrated in Figure 6.2. It can be observed that the satellite is located above the Netherlands, while the target is positioned in Rome. Additionally, the illumination conditions correspond to the expected transition between nighttime and natural light during nautical twilight. This confirms that the geodetic initialization and solar geometry are correctly represented within the simulation.

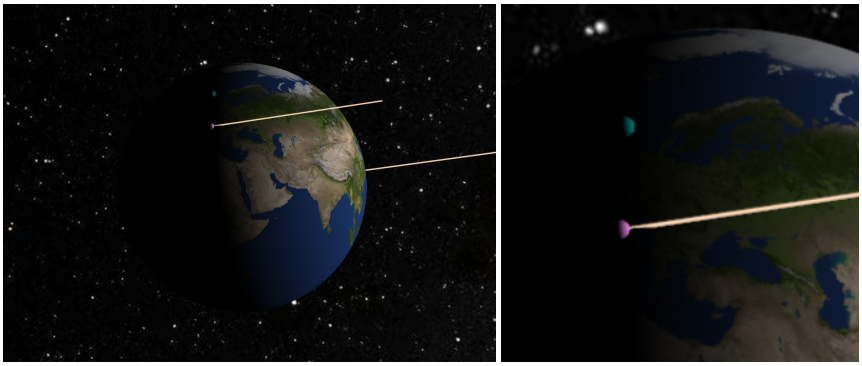


Figure 6.2: Placement of the satellite above Amsterdam and the target in Rome on 21 June 2025 during nautical twilight. It can be observed that the satellite and target are positioned correctly and that the illumination conditions match the expected lighting transition.

Orbital Period

The orbital propagation of the satellites can be validated by analytically computing the orbital period and comparing this value with the result obtained from a simulation run.

The orbital period of a satellite in a Keplerian orbit is given by Equation 6.1. Using the orbital parameters defined in Table 5.1, corresponding to a WorldView-3-like satellite with an apogee altitude of $624.6\,\mathrm{km}$ and a perigee altitude of $615.7\,\mathrm{km}$, the orbital period can be determined by substituting the parameters into the analytical formulation in Equation 6.1.

$$T = 2\pi\sqrt{\frac{a^3}{\mu}} = 2\pi\sqrt{\frac{\left(6371 + \frac{624.6+615.7}{2}\right)^3}{398600}} = 5820.38\ \mathrm{s} = 97.01\ \mathrm{min} \tag{6.1}$$

Here, a denotes the semi-major axis of the orbit and μ represents the Earth's gravitational parameter. Substituting the orbital parameters yields an orbital period of approximately $5820\,\mathrm{s}$.

To verify this result, a simulation of 24 hours was executed using the same orbital parameters. From the simulation logs it was observed that the satellite completed 14.83 orbits during this time interval. Dividing the total simulation time by the number of completed orbits results in

$$\frac{24 \times 3600}{14.83} = 5826.4 \text{ s}$$

This value matches the analytical orbital period derived from Equation 6.1. Therefore, the agreement between the analytical result and the simulated orbit count confirms the correct implementation of the Orekit orbital propagation within the simulation framework.

Target Propagation

The target propagation model was verified to ensure that targets remain confined to ocean regions and do not cross coastline boundaries. To test this behavior, a simulation was conducted using accelerated time while increasing the target movement speed. The resulting motion was recorded and is available at: https://youtu.be/FtsqwnMiFVs.

An example screenshot is shown in Figure 6.3. Although the figure provides a static snapshot, it illustrates that the targets remain within ocean regions and do not cross coastal boundaries. Furthermore, the trajectories appear smooth and continuous, indicating that the propagation model produces physically consistent movement over the ocean surface.

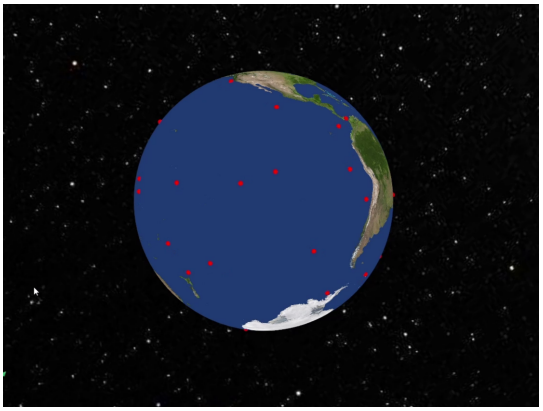


Figure 6.3: Screenshot of the visualization of target propagation in accelerated time. Although static, the figure shows that the targets remain confined to ocean regions and do not cross coastal boundaries.

From a ground projection as shown in Figure 6.4, relative to the field of viewing window of the satellite of 13.1 km, it can be derived that the whale covered around 30 km within 4 hours time, corresponding to 7.5 km / h. This is in the order of the average speed of 1.5 * 3.6 = 5.4 km/h, with an upper bound of 6.0 m/s = 21 km / h, therefore within the requested input bounds, confirming a representative implementation of the target dynamics.

It can also be observed that, given the limited displacement of the whale, it is highly unlikely that it will move out of view in the orders of the Tip and Cue latencies. For the propagation speed to become relevant, the target would need to cover distances on the order of kilometres within the 5 minute Tip–Cue delay, to increase chances to move out the 13.1 km swath view. In the order of 1-2 km in five minutes time delay, gives average moving speeds of above 60 km/h for it to become relevant. For marine targets, this is very unlikely. This is even below the speed of most commercial ships.

Instead, in the case of whales, the effect of submerge after 10 minutes would be have been more interesting to model rather than its propagation speed for out of view. Still, when higher detailed observations are desired, the viewing windows may even become smaller given the EO trade off of spatial resolution vs field of view. If the window becomes smaller, and at higher speed applications like vessels, then the effect of moving targets may become relevant again.

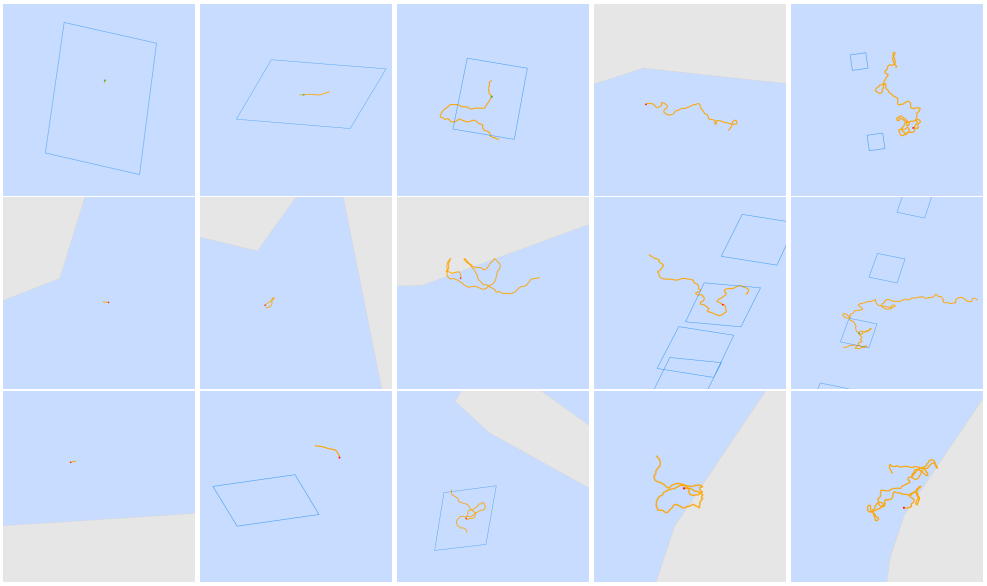


Figure 6.4: Example target movement over time. From left to right the positions after 5 minutes, 30 minutes, 4 hours, 8 hours, and 24 hours are shown, from which the distance with respect to the 13.1 swath window can be estimated.

6.1.2. Observation Geometry

Field of View Projection

The field of view projections were verified by specifying a satellite and target above the center of Gran Canaria at latitude $27.94716°$ and longitude $-15.60062°$, and by initiating a nadir pointing request exactly to that location. The field of view was set to $45\,\mathrm{km}$ to match the diameter of Gran Canaria [169].

The result is shown in Figure 6.5. It can be confirmed that the boresight ray exactly intersects the center of the target window. In addition, by matching the outer diameter of Gran Canaria, it can be observed that the outer field-of-view points were also placed correctly. This confirms the correct pointing implementation and provides an internal validation of the reference frame transformations.

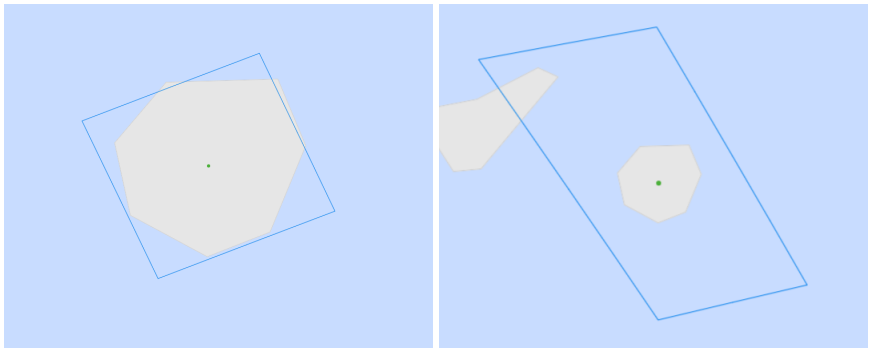


Figure 6.5: Verification of the field-of-view projection above Gran Canaria. Left: nadir pointing where the footprint is centered on the island. Right: off-nadir pointing after relocating the satellite, showing elongation of the footprint.

The nadir corner points of the field-of-view window are given in Table 6.1.

Corner	Latitude (°)	Longitude (°)
Top	28.21061	-15.48147
Left	28.05236	-15.89763
Bottom	27.68358	-15.71915
Right	27.84129	-15.30415

Table 6.1: Correct Corner Coordinates

Next, the off-nadir transformations were verified. To do so, the satellite was arbitrarily moved from longitude -15.60062 to -10.60062 and from latitude 27.94716 to 20.94716. This produces an off-nadir observation with angle $\theta = 53.502°$, which is also shown in Figure 6.5. The new coordinates of the field-of-view points are given in Table 6.2.

Corner	Latitude (°)	Longitude (°)
Top	28.78108	-15.61096
Left	28.60727	-16.56446
Bottom	27.21598	-15.60338
Right	27.40303	-14.80598

Table 6.2: Corner Coordinates

To quantify the geometric change in footprint size, the ground distance between the top and bottom footprint points was computed using the haversine formula [170]. For two geodetic coordinates (φ_1, λ_1) and (φ_2, λ_2), the ground distance is

$$L = 2R_{\oplus} \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos\varphi_1 \cos\varphi_2 \sin^2\left(\frac{\Delta\lambda}{2}\right)}\right), \tag{6.2}$$

where $\Delta\varphi = \varphi_2 - \varphi_1$ and $\Delta\lambda = \lambda_2 - \lambda_1$.

Substituting the nadir top and bottom coordinates from Table 6.1, $(\varphi_1, \lambda_1) = (28.21061°, -15.48147°)$ and $(\varphi_2, \lambda_2) = (27.68358°, -15.71915°)$, gives $L_{\text{nadir}} \approx 63.08\,\text{km}$. Repeating the same computation for the off-nadir top and bottom coordinates from Table 6.2, $(28.78108°, -15.61096°)$ and $(27.21598°, -15.60338°)$, gives $L_{\text{off}} \approx 174.03\,\text{km}$. The measured enlargement of the footprint is therefore

$$\frac{L_{\text{off}}}{L_{\text{nadir}}} = \frac{174.03}{63.08} \approx 2.76. \tag{6.3}$$

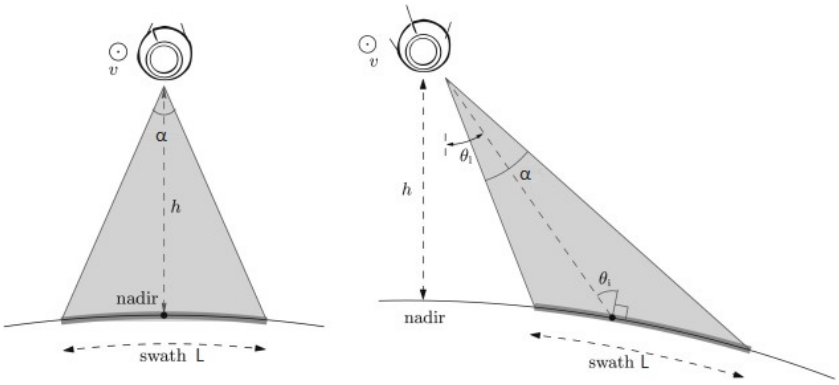


Figure 6.6: Geometry of the sensor footprint projection, illustrating the relation between satellite altitude h, off-nadir angle θ, field-of-view angle α, and footprint length L. [171]

Next, it is verified whether this measured enlargement is consistent with the sensor-footprint geometry illustrated in Figure 6.6. For satellite altitude h, off-nadir angle θ, and full field-of-view angle α, the footprint length in the viewing direction is given by Equation 6.4.

$$L(\theta) = h\left[\tan\left(\theta + \frac{\alpha}{2}\right) - \tan\left(\theta - \frac{\alpha}{2}\right)\right]. \tag{6.4}$$

For nadir pointing this reduces to Equation 6.5.

$$L(0) = 2h \tan\left(\frac{\alpha}{2}\right). \tag{6.5}$$

The predicted enlargement factor therefore becomes Equation 6.6.

$$\frac{L(\theta)}{L(0)} = \frac{\tan\left(\theta + \frac{\alpha}{2}\right) - \tan\left(\theta - \frac{\alpha}{2}\right)}{2\tan\left(\frac{\alpha}{2}\right)}. \tag{6.6}$$

Using the simulation parameters $h = 613\,\mathrm{km}$ and swath width $45\,\mathrm{km}$, the field-of-view angle obtained from Equation 6.5 is

$$\alpha = 2\arctan\left(\frac{45}{2 \cdot 613}\right) \approx 4.204^\circ. \tag{6.7}$$

Substituting $\theta = 53.502^\circ$ and $\alpha = 4.204^\circ$ into Equation 6.6 yields

$$\frac{L(\theta)}{L(0)} \approx 2.83. \tag{6.8}$$

The measured enlargement factor of 2.76 agrees closely with the analytical approximation of 2.83. The relative difference is approximately $\frac{2.76-2.83}{2.83} \times 100\% \approx -2.5\%$. This deviation arises because Equation 6.4 assumes a planar ground geometry, while the simulation computes the exact intersection of the viewing rays with the curved Earth ellipsoid. At large off-nadir angles such as $\theta = 53.502^\circ$, Earth curvature modifies the projected footprint length compared to the planar approximation, which matches expected deviations [172].

The field-of-view projections were also plotted for a full orbital period, as shown in Figure 6.7. This shows the expected propagation behavior over time, where after one full orbital cycle, the longitude of the starting and ending window coincides.

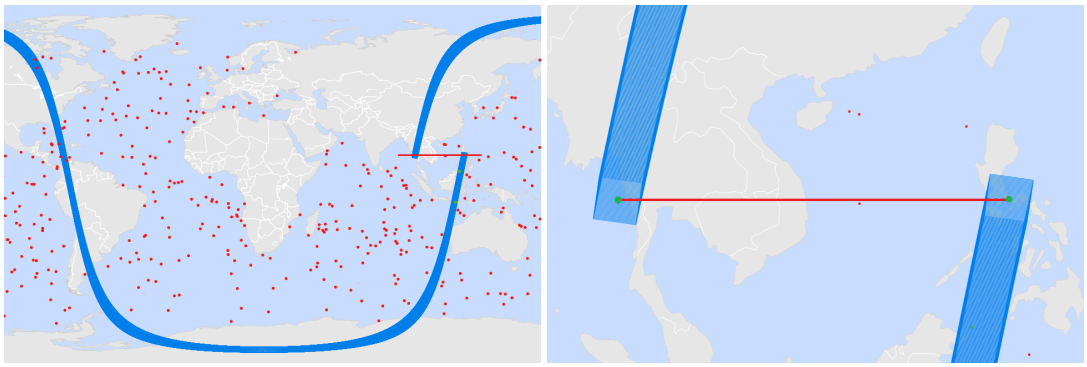


Figure 6.7: Footprints of one simulation run for one full orbit. It can be observed that the footprints coincide in latitude after one orbital cycle, confirming correct orbital propagation.

Time Delay

The time delay between the Tip and Cue satellite was validated by plotting their field-of-view projections and continuing the simulation for the duration of the time delay between Tip and Cue. After that propagation, both satellites should have completed the same orbital distance, and the Cue satellite should have reached exactly the location where the Tip satellite started initially.

The result is shown in Figure 6.8. It can be observed that the Cue window at its end position aligns with the Tip window at its initial position, verifying that the time delay has been set correctly.

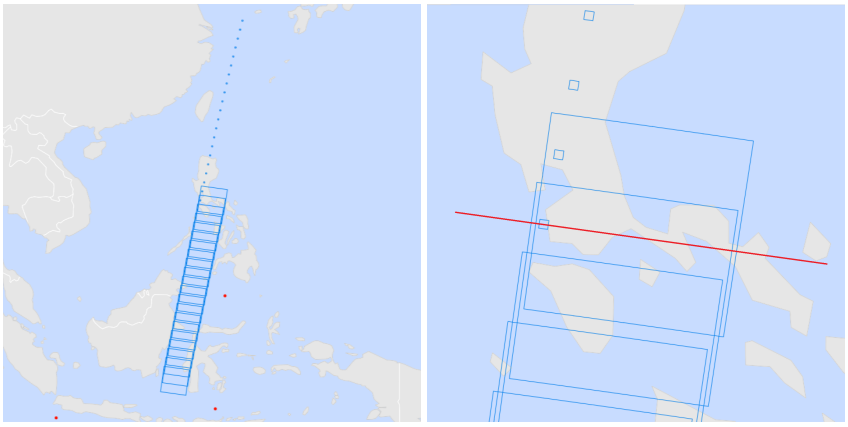


Figure 6.8: Verification of the time delay between Tip and Cue satellites. After propagating for the configured delay, the Cue footprint aligns with the initial Tip footprint position.

6.1.3. Tip and Cue Tasking Logic

Illumination Constraints

Given that the radiometric effects are heavily dependent on lighting conditions, it is important to thoroughly verify the solar position implementation.

First, the solar cycle was verified by setting the time to 4:00, 8:00, 12:00, 16:00, 20:00, and 0:00 CET on 21 March 2025. The result is shown in Figure 6.9, where the position of the Sun points towards its expected location and the targets located in darkness were filtered out accordingly.

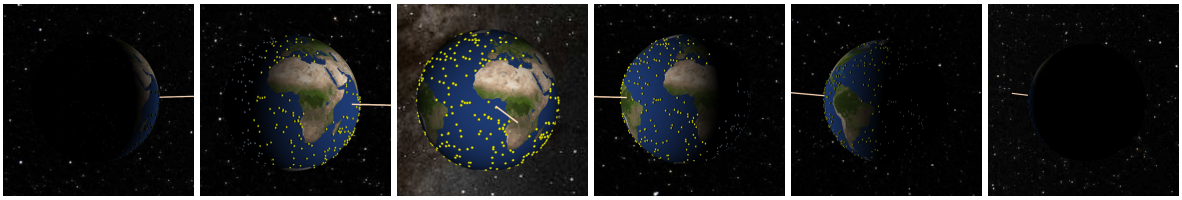


Figure 6.9: Sun-Earth geometry at different UTC times on 21 March 2025. The figures show the expected movement of the sun direction over the course of the day.

After that, the Earth tilt was verified by simulating the peak dates of all equinoxes and solstices: 21 March, 21 June, 21 September, and 21 December. The lighting was set to 12:00 for all cases, not corresponding to the exact equinox or solstice times, but sufficient for visualization purposes. Figure 6.10 shows the resulting illumination conditions. Examining the overall lighting conditions, for example in the Arctic regions, highlights the polar days and nights expected around these times, indicating that the Sun direction is set correctly.

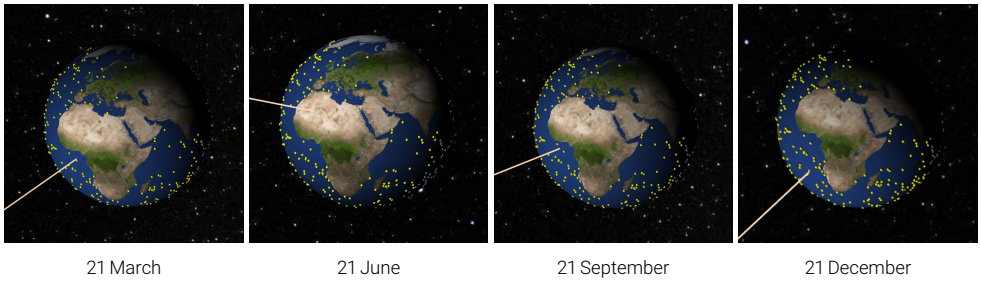


Figure 6.10: Sun–Earth geometry at different seasons on 21 March (spring), 21 June (summer), 21 September (autumn), and 21 December (winter) 2025. The illumination patterns illustrate the expected seasonal tilt of the Earth.

Next, all satellite and target objects were plotted to verify whether the mask was applied correctly. The result is shown from different viewing perspectives in Figure 6.11, where it can be observed that both satellites and targets were either included or filtered out correctly.

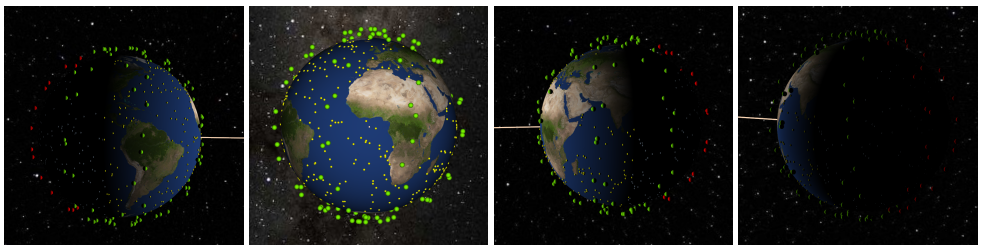


Figure 6.11: Visualization of the illumination mask applied to satellites and targets. From the different perspectives it can be observed that objects in darkness are correctly filtered out.

Tasking

The tasking logic was validated by modelling many targets in close proximity and visualizing in which order the tasks were evaluated. To do so, a case with 10,000 targets was simulated, as shown in Figure 6.12. A simulation video is available at: https://youtu.be/nxt2wQQKZeQ .

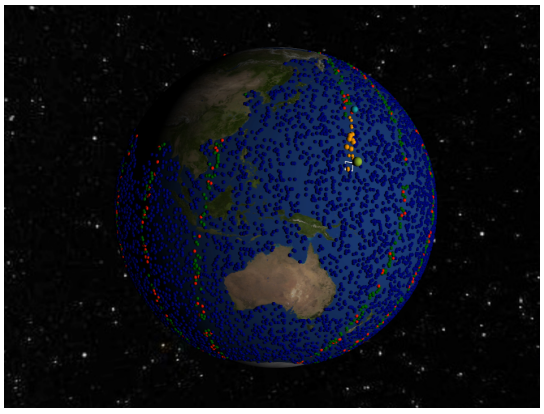


Figure 6.12: AI-based Tip and Cue simulation with 10,000 targets used to verify the tasking logic implementation.

A footprint overview, including which targets have been evaluated, is shown in Figure 6.13.

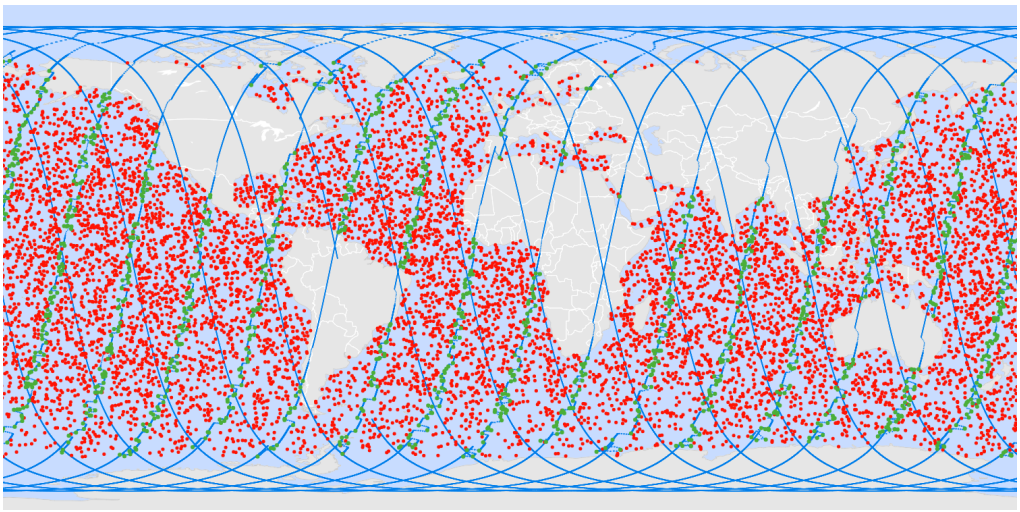


Figure 6.13: Cue satellite footprints for an AI-based Tip and Cue mission with 10,000 targets. Observed targets are shown in green, while unobserved targets remain red.

Zooming in, Figure 6.14 shows the order of a basic forward-pointing tasking operation. After receiving a task from the Tip satellite and stabilizing, the Cue satellite observes Target 1. Next, while receiving the next task before approaching that target, it prepares its stabilization in advance for Target 2. In the same way, it observes Target 3 after that. After completing the full task list, it returns to nadir position, indicated by the letter N, which can be recognized by the narrow nadir field-of-view projections. This evaluation sequence matches what the cost function is optimizing for, namely a minimal Euler angle change with respect to the current position, for all tasks that can still be reached within the viewing window.

Next, a more complex operation involving backward pointing was also evaluated. It can be observed that the Cue satellite points more backward than strictly required, waits for the required stabilization time, observes the target, and then continues to the next task. This matches the stabilization controller described in subsection 5.2.4.

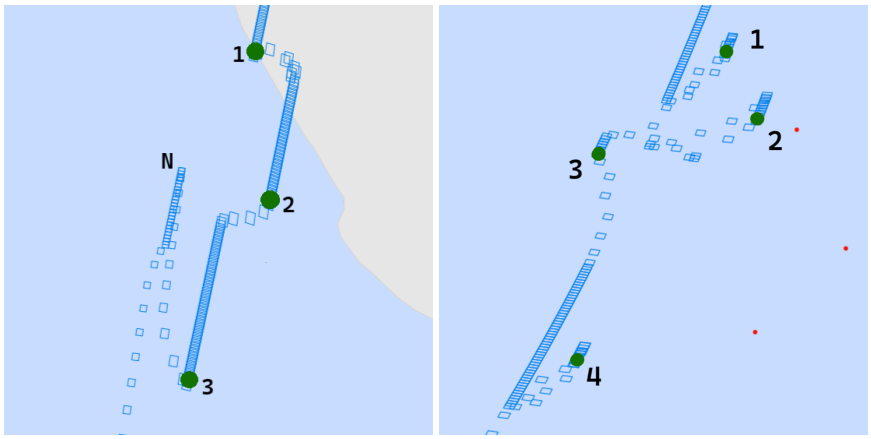


Figure 6.14: Examples of Cue satellite tasking operations. Left: forward-pointing tasking sequence. Right: backward-pointing maneuver showing stabilization before observation.

Sequential pointing operations are shown in Figure 6.15. From the left figure, it can be seen that some targets near the Cue satellite are not observed, despite being on the track and therefore being observed by the Tip. This is because the modelled onboard detection model on the Tip satellite, with a TPR and TNR of 0.85, falsely negatively classified some targets as non-whales. Therefore, they were not passed on to the tasking list of the Cue. Alternatively, the Cue satellite may have optimized its observations for targets that could still be reached in time, meaning that the missed targets did not allow for an optimal observation within the time constraints.

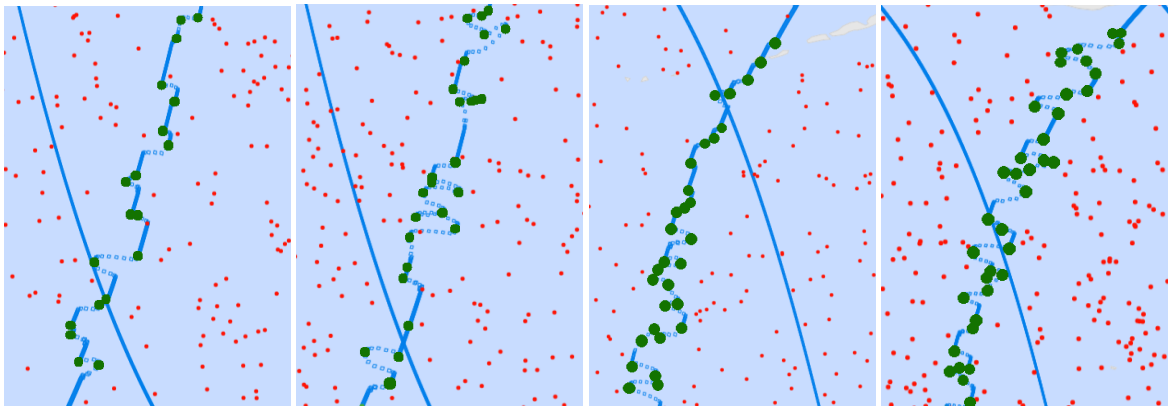


Figure 6.15: Forward tasking operations for multiple targets. Some targets within the Tip footprint are not observed because they were falsely classified by the Tip satellite or could not be reached within the available time window.

The order of task evaluation is shown in Figure 6.16. It can be observed that tasks 1–11 are evaluated in a logical order based on the cost function for which the system is optimized, namely minimizing the

time to reach with respect to each current position. With that, it evaluates clusters 1–5 in sequential order, 6–10 in sequential order, and then jumps back to 11 to evaluate the last point. This confirms a correct implementation, also noting that task 11 was received earlier than task 7 because it appeared earlier in the Tip satellite footprint.

A direct path operation is also shown in Figure 6.16. It confirms that the Cue satellite indeed evaluates the tasks that require the smallest time to reach. This is an instantaneous optimization without forward-looking capability, where, for example, target 12 could have been evaluated before target 11 to provide better future reachability. However, tasking optimization falls outside the scope of this work, as it also depends on other factors such as the target interest score that follows from the Tip, which could be made more advanced using LLM-based target descriptions and scoring. Given the modular tasking implementation, this work is continued by next ESA ϕ-lab visiting researchers building on the AI-Based Tip and Cue simulation framework.

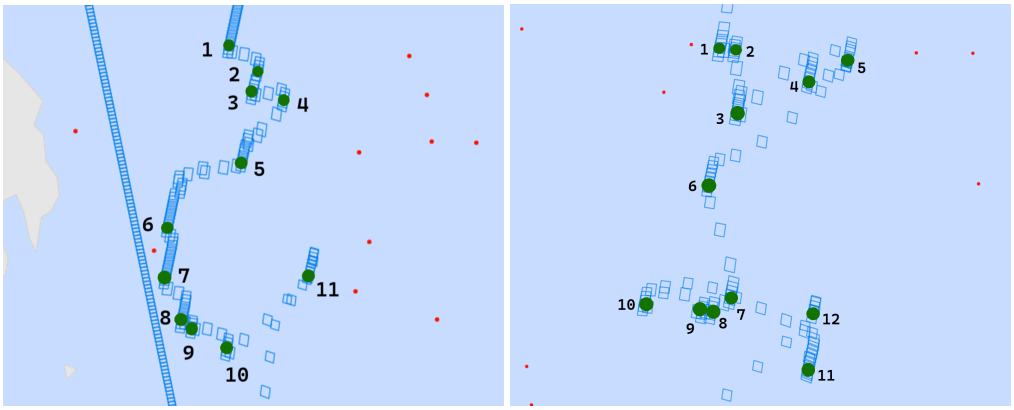


Figure 6.16: Task evaluation order of the Cue satellite for targets 1-11. The sequence illustrates that tasks are evaluated based on the minimum time-to-reach according to the cost function.

As this shows that all tasks are handled within the given time, in mostly logical order, and that the latency case for 500 targets rather than 10,000 will mostly involve individual cases, this tasking implementation provides an accurate estimate of latency and remaining viewing time per target. It therefore satisfies its intended purpose within the scope of this work.

It should be noted that in practice, given realistic processing time, it would not be possible to process and confirm this many targets within such close proximity. Therefore, this configuration should only be used for tasking verification purposes.

6.1.4. Pointing and Stabilization

The pointing and stabilization planning was verified by plotting the viewing window over time and comparing it to the computed stabilization times.

Figure 6.17 shows a stabilization maneuver. Visually, it can be observed that the windows are initially close to each other, then come further apart in the acceleration phase, after that are again at constant spacing during the constant-velocity phase, and finally come closer together in the deceleration phase. This is as expected, because the Euler angle change follows the slewing behavior described in subsection 5.2.4, which confirms the implementation of the maneuver dynamics.

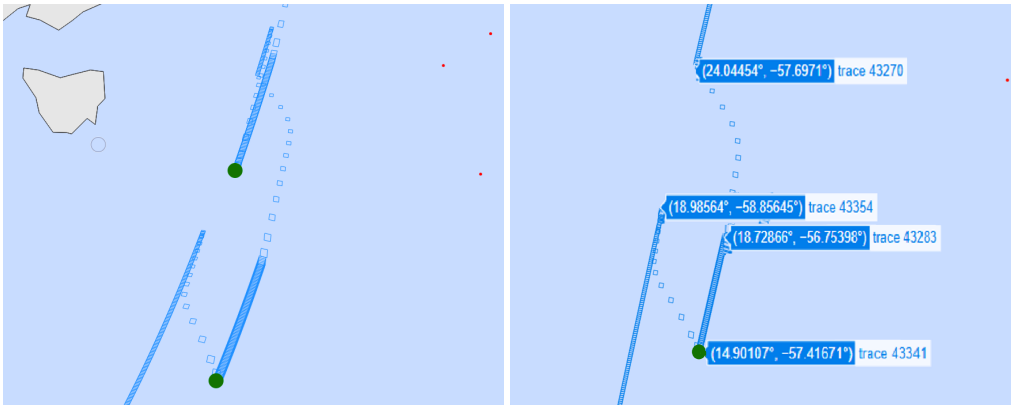


Figure 6.17: Example stabilization maneuver. The changing spacing between footprints reflects the acceleration, constant-rate, and deceleration phases of the slewing motion.

Next, the slewing time was analytically verified. Observing a single-target operation in Figure 6.17, the first move from nadir to a stabilized off-nadir position starts at frame 43270 and ends at frame 43283. The second move back to nadir starts at frame 43341 and ends at frame 43354. With a step size of 1 second between the frames, this leads to a maneuver duration of 13 seconds for both slews. The durations are expected to be similar, given that the satellite moves from nadir to position and from position back to nadir.

Using Equation 5.34, the slewing time can be evaluated analytically. From the logs, the target at latitude 14.96 and longitude −57.5 was observed at roll, pitch, yaw angles of $(-17.17573843,\ -36.6885644,\ 5.733309084)°$. Starting from nadir pointing $(0, 0, 0)°$, the largest required maneuver occurs in pitch with a change of $\Delta\theta = 36.6885644°$. Taking the maximum angular velocity and acceleration of 3.86 and 1.43 respectively from the mission, similar to WorldView-3 [173], Equation 5.32 from subsection 5.2.4 gives the individual maneuver phases in Equation 6.9 and Equation 6.10.

$$t_{acc} = t_{dec} = \frac{\omega_{max}}{\alpha_{max}} = \frac{3.86}{1.43} = 2.70 \text{ s} \tag{6.9}$$

$$t_{const} = \frac{\Delta\theta - \frac{\omega_{max}^2}{\alpha_{max}}}{\omega_{max}} = \frac{36.6885644 - \frac{3.86^2}{1.43}}{3.86} = \frac{36.6885644 - 10.419}{3.86} = 6.81 \text{ s} \tag{6.10}$$

Substituting these results into Equation 5.32, the total slewing time is evaluated with Equation 6.11.

$$t_{slew} = t_{acc} + t_{const} + t_{dec} = 2.70 + 6.81 + 2.70 = 12.21 \text{ s} \tag{6.11}$$

This results in a total slewing time of approximately 12.2 seconds, which matches the 13-second estimate obtained from the frame dynamics. This value is also representative of values from literature, where WorldView-3 requires 10 seconds to slew 200 km [173].

Because the target falls perfectly in the middle of the observation window after completing the slew, this confirms that the forward slewing dynamics have been implemented correctly in the simulation framework.

6.1.5. Visual Validation

Having defined the full simulation framework, a visual validation of an entire mission simulation was conducted. This shows the full propagation of all satellites and lighting over time, including their moving field-of-view windows and target states.

The result is automatically recorded for every simulation. For the sake of this report, it is made available on YouTube for five distinct mission configurations: a single Tip and Cue system, a Tip and Cue constellation,

and a VHR constellation, independently operating satellites, and 10,000 targets, for which the links are provided in Table 6.3. Screenshots for each respective simulation are attached in Figure 6.18.

Table 6.3: Simulation Video Links for the Evaluated Satellite Configurations

Configuration	Video Link
Single Tip and Cue System	https://youtu.be/RVM3C2Z773Q
4x4 Walker Delta Constellation	https://youtu.be/uwNv4IQcf1I
8x2 Walker Delta VHR Constellation	https://youtu.be/GwfU0NNwcBk
Independently Operating Satellites	https://youtu.be/AJPzTOfFrVA
Single Tip and Cue System, 10,000 Targets	https://youtu.be/nxt2wQQKZeQ

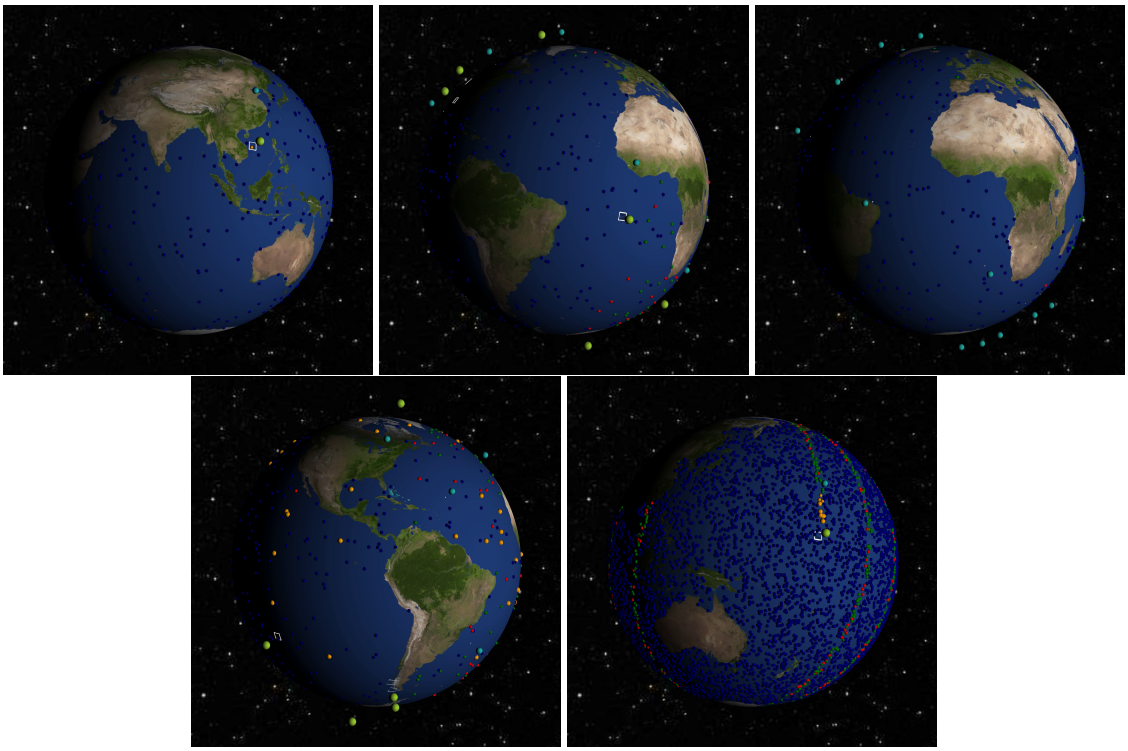


Figure 6.18: Examples of full mission simulations. From left to right: single Tip and Cue formation, Tip and Cue Walker Delta constellation, VHR Walker Delta constellation, independently operating Tip and Cue, and single Tip and Cue system with 10,000 targets.

From the propagated videos, it can be observed that the target states are tracked correctly: after a Tip observation, targets turn orange; after the Tip processing time when they are tasked, their size increases, after a Cue observation, the size decreases again, and finally, after the Cue processing time has passed, the colour changes to green (positive) or red (negative) confirmation.

In addition, the Cue satellites are tasked correctly. This is especially evident in the independent operating satellite case, where the Cue satellites handle tasks upon overpass.

A complete footprint log from a 24-hour simulation for both the Tip and Cue satellite is shown in Figure 6.19 and Figure 6.20, respectively. This confirms that the targets are only logged in the observed state, with the green observed targets only falling within the satellite footprints. At the same time, it can be observed that not all targets within the Tip satellite footprint reach the observed state, as some of them remain red. This is to be expected, as targets that fall within nighttime are automatically filtered out and cannot be observed. However, that effect is not directly visible within the footprint plot itself.

The Cue pattern in Figure 6.20 matches the orbital tracks of the Tip, which is expected because the Cue follows the Tip. From a pointing and stabilization perspective, it correctly reaches every target at off-nadir

positions through a stable sequence of field-of-view patterns. While the observation of smooth movement alone is not sufficient to confirm correctness, the verification for pointing and stabilization has already been given in subsection 6.1.4.

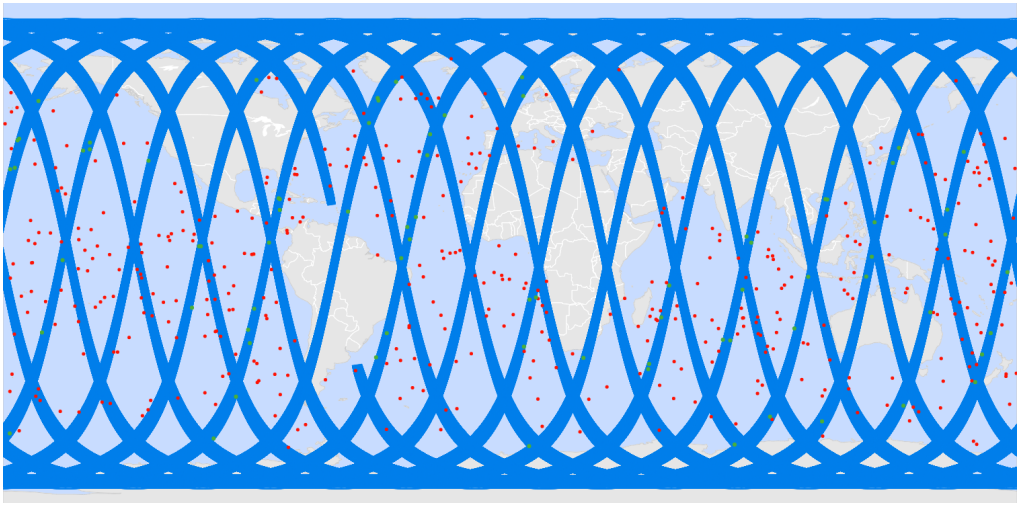


Figure 6.19: Twenty-four hour footprint log of the Tip satellite. Observed targets in green appear only within the Tip satellite footprints.

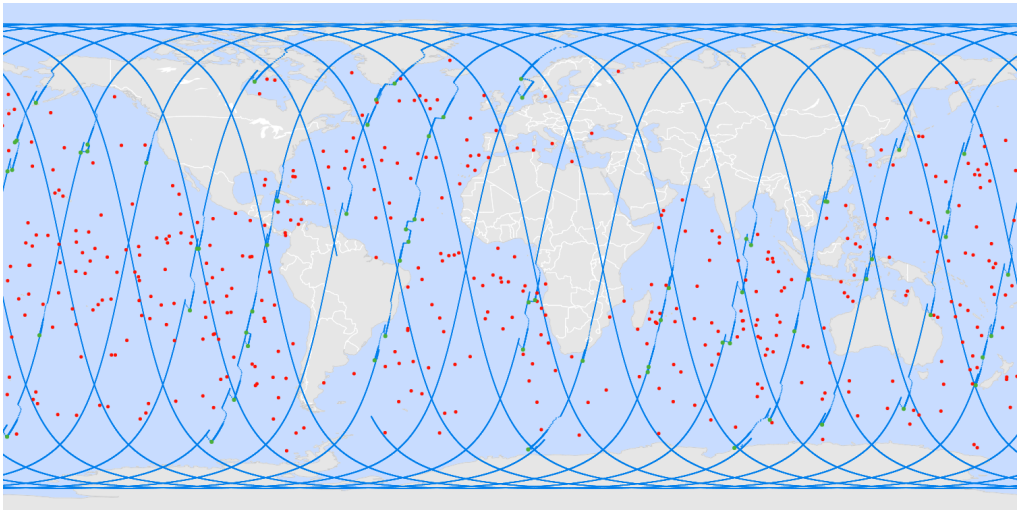


Figure 6.20: Twenty-four hour footprint log of the Cue satellite. The Cue footprints follow the Tip orbit while executing off-nadir observations of tasked targets with smaller coverage, but higher spatial resolution.

6.1.6. Data Logging and State Handling

To verify whether the targets are logged correctly, the observations were traced back in time, where the simulation time is displayed in the lower corner and verified through the simulation logs. Example detections were taken from the simulation video for "$TC\_1x1sat\_40deg\_5min\_42sd$", corresponding to a Tip and Cue simulation with a single Tip and Cue satellite, a $40°$ maximum off-nadir requirement, a 5-minute Tip-Cue delay, and target seed 42, as shown in Figure 6.21, Figure 6.22, and Figure 6.23.

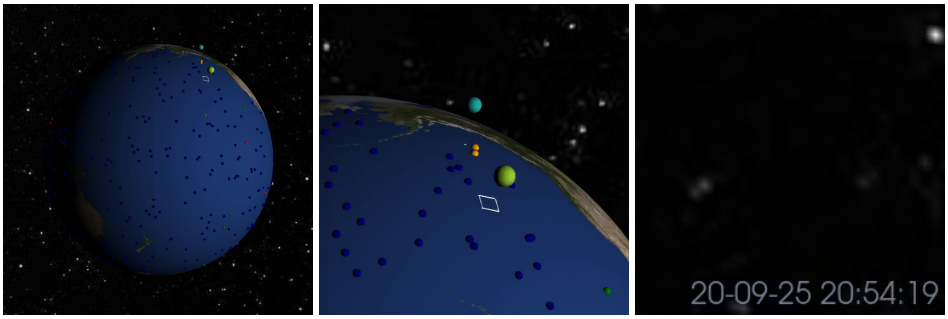


Figure 6.21: Observation example showing the global overview (left), a zoomed view around the observation footprint (center), and the simulation time (right) for the first detection event.

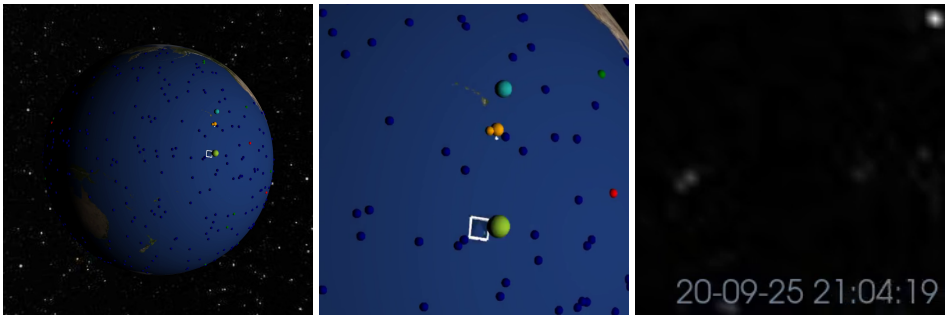


Figure 6.22: Observation example showing the global overview (left), a zoomed view around the observation footprint (center), and the simulation time (right) for the second detection event.

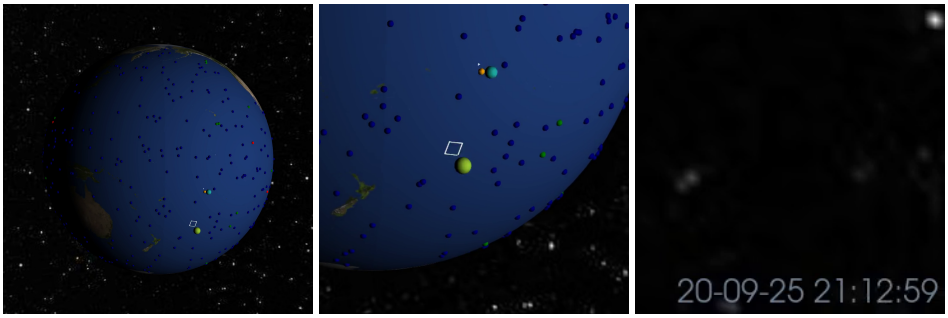


Figure 6.23: Observation example showing the global overview (left), a zoomed view around the observation footprint (center), and the simulation time (right) for the third detection event.

Tracing back the logs after around 20:54 gives the result in Table 6.4.

Table 6.4: Extract of cue detections from $results\_TC\_1x1sat\_40deg\_5min\_42sd.xlsx$ file.

Detection ID	Target ID	Cue Time	Target Lat	Target Lon	Target Alt	Cue Lat	Cue Lon	Cue Alt
...44dc30	435	2025-09-20T20:54:41Z	23.2344	53.0627	-142.7103	58.0057	-143.0654	623793.9939
...22767b	333	2025-09-20T20:55:13Z	5.8896	51.1835	-144.7281	56.0929	-144.1171	623173.7064
...4182c2	292	2025-09-20T21:04:59Z	23.5431	15.4633	-155.5236	20.3669	-155.5117	613033.7583
...f5a657	59	2025-09-20T21:06:36Z	27.8907	15.6536	-154.3521	14.3911	-156.8275	612234.0081
...744a85	179	2025-09-20T21:13:03Z	23.5689	-14.3991	-161.8367	-9.4854	-161.8059	613296.9006

Indeed, the times match the simulation time, and the coordinates also correspond to the detection locations within the North and South Pacific Ocean. As these are inputs that will be passed on to the off-nadir image generation pipeline in section 5.3, this confirms that the observations are logged correctly.

After confirming this final step, namely the observation logging, the validation of the AI-Based Tip and Cue orbital simulation framework is completed.

6.2. Off-Nadir Imaging

This section verifies and validates the off-nadir imaging pipeline developed in this work. The objective is to assess whether the implemented sensor geometry, radiometric modelling, and reflectance generation behave consistently with the physical image-formation process. The pipeline is therefore evaluated at multiple levels, starting from controlled internal checks of projection geometry and radiometric consistency, and ending with comparison against real Φ-Sat-2 observations. In that way, it can be determined whether the rendered off-nadir images provide a physically representative basis for the onboard detection experiments in Chapter 7.

6.2.1. Sensor Geometry

To verify if the sensor geometry has been defined correctly, the projection is verified by rendering the texture of an image captured nadir, which should be exactly the same as the original input image.

A colored checkerboard image with an orientation mark was loaded into the renderer onto a DEM surface with zero elevation (`wave_min = wave_max = 0`). The satellite and target were configured so that the off-nadir angle is $0°$, meaning that the viewing direction is aligned with the local surface normal. The rendered texture (constant illumination, no radiance model) is then compared directly against the original input image.

Figure 6.24 shows the result. Visually, the projected texture aligns with the original: the checkerboard squares and the orientation mark overlap without a systematic shift, rotation, or scaling error. The absolute-difference images show only sparse, small-amplitude deviations concentrated at high-contrast edges (checker boundaries and the orientation mark). This behaviour is expected for a Monte Carlo renderer with finite samples, where small stochastic fluctuations and subpixel sampling at sharp discontinuities cause slight pixel-value differences even when the underlying geometry is correct.

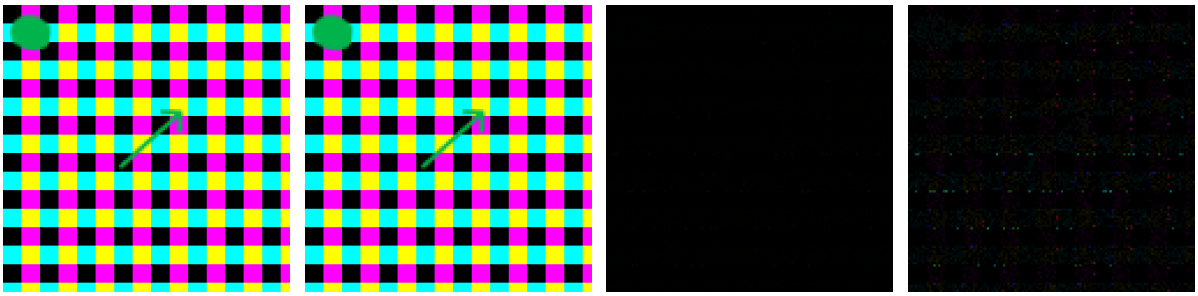


Figure 6.24: Sensor-geometry verification at $0°$ off-nadir using a flat DEM and a checkerboard texture. From left to right: original RGB image (input), rendered nadir reprojected texture under constant illumination (output), absolute difference $|\Delta|$ (linear scale), and $|\Delta|$ amplified by a factor 10 for visibility. The absence of a coherent shift pattern confirms correct camera geometry and UV mapping; remaining differences occur at high-contrast edges due to finite-sample variance and subpixel sampling.

6.2.2. Sample Count

After verifying that the nadir projection reproduces the input texture geometry, the remaining deviations are mainly attributed to Monte Carlo variance and rasterisation/sampling effects at discontinuities. To verify that the selected sample count is sufficiently high, the same nadir texture render (subsection 6.2.1) is repeated for increasing samples-per-pixel (SPP), and the absolute difference between the rendered texture and the original input is evaluated.

Two error measures are reported: the mean absolute difference over all pixels, and the maximum absolute difference over all pixels. Both are computed for the full RGB image, taken as an absolute value per pixel, and per channel R/G/B. In addition, the wall-clock render time per image is recorded to quantify the computational cost of increasing SPP.

Figure 6.25 shows the mean absolute difference as a function of sample count. The mean error decreases quickly between 2^7 and 2^{11} samples per pixel, where it reduces from approximately 0.66 DN to about

0.32 DN. Around 2^{12} samples per pixel, the mean reaches its minimum of approximately 0.315 DN, after which it remains nearly constant and even increases slightly for higher sample counts. This behaviour indicates that the Monte Carlo variance has largely converged and the remaining residual is dominated by deterministic effects like subpixel interpolation and texture filtering at sharp discontinuities.

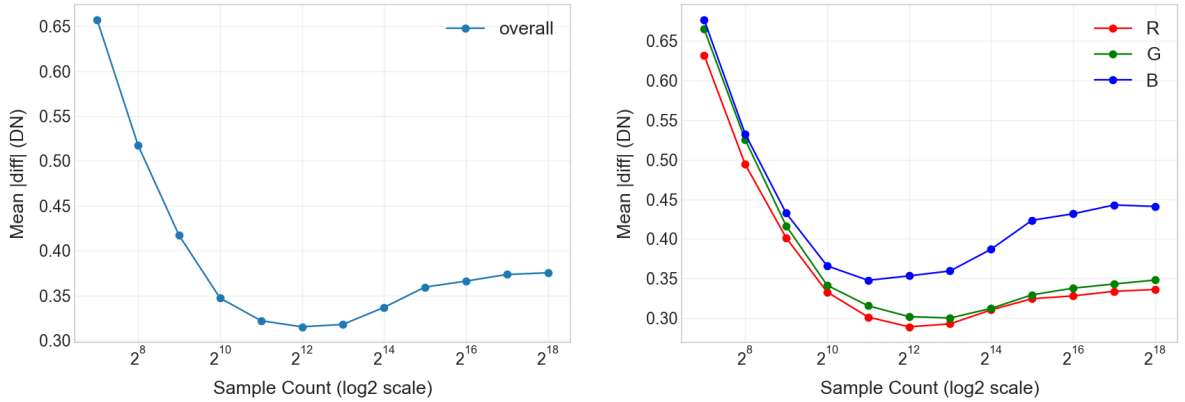


Figure 6.25: Mean absolute difference between the rendered texture and the original input as a function of sample count ($\log_2$ scale on the x-axis). Left: overall mean absolute difference over RGB. Right: mean absolute difference per band. The mean error decreases rapidly at low sample counts and then saturates, indicating that residual differences are dominated by a small set of edge pixels and quantisation/sampling effects rather than bulk misprojection.

The per-band curves follow a similar trend. The blue channel shows slightly higher residual mean differences than red and green. This is consistent with the spectral distribution of the input image itself: the normalized channel energy is $R = 0.325$, $G = 0.334$, and $B = 0.340$, meaning that the image contains slightly more blue intensity overall. Since the absolute difference is measured in digital numbers (DN), channels with higher overall magnitude and stronger high-frequency transitions contribute to slightly higher residual errors. The effect, however, remains small compared to the total reduction achieved by increasing SPP.

Figure 6.26 shows the maximum absolute difference as well. The maximum error decreases monotonically from 33 DN at 2^7 samples per pixel to 4 DN at 2^{17} samples per pixel, and reaches 3-4 DN at the highest tested sample count of 2^{18}. A flattening is visible around 2^{14} samples per pixel, beyond which the reduction in maximum error becomes progressively smaller. This indicates that the remaining maximum deviations are dominated by a small number of pixels at strong discontinuities, like the checker edges and the orientation mark.

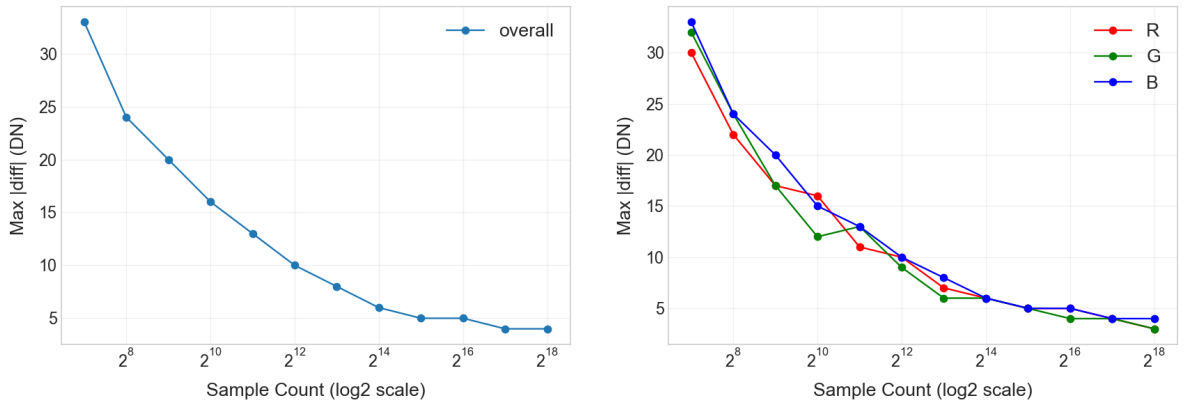


Figure 6.26: Maximum absolute difference between the rendered texture and the original input as a function of sample count ($\log_2$ scale on the x-axis). Left: overall maximum absolute difference over RGB. Right: maximum absolute difference per band. The maximum error decreases monotonically and shows decreasing returns at high sample counts, where remaining outliers are concentrated at sharp discontinuities.

Finally, Figure 6.27 shows the render time per image. Runtime increases strongly for higher sample counts, reaching 12.5 seconds at 2^{18} samples per pixel. In contrast, both the mean and maximum error metrics saturate beyond approximately 2^{12}-2^{13} samples per pixel. This provides a practical operating point for the pipeline: sample counts significantly above this point increase computational cost while improving the texture reprojection accuracy only marginally. In the remainder of this work, the selected sample count is therefore chosen as 2^{12} to balance convergence and computational feasibility.

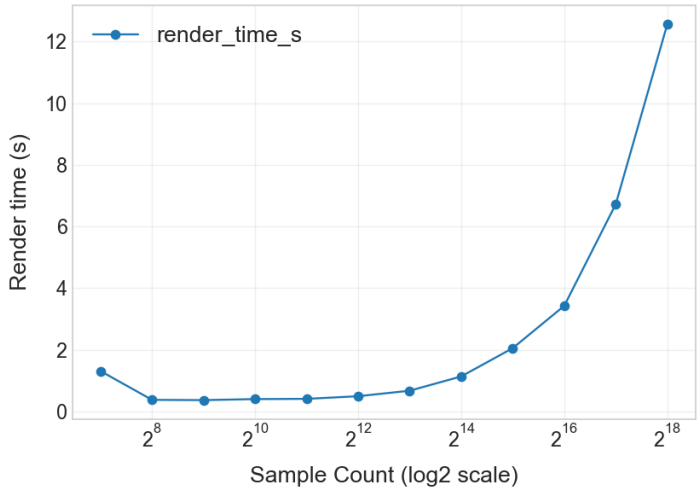


Figure 6.27: Measured render time per image as a function of sample count ($\log_2$ scale on the x-axis). Runtime increases strongly with sample count, while the error metrics in Figure 6.25 and Figure 6.26 saturate, motivating the selection of a mid-to-high sample count in the knee region as a trade-off between accuracy and computational cost.

6.2.3. Radiative Transfer Model Representation

To validate the radiative transfer model, the generated radiative transfer curves are compared to examples from literature [174]. Both the direct solar spectrum and the diffuse sky spectrum are evaluated.

A SMARTS-generated spectrum is shown next to the literature example in Figure 6.28. The literature reference distinguishes three curves: the extraterrestrial spectrum, the direct beam at the surface, and the total irradiance on a horizontal surface. The extraterrestrial spectrum represents the solar spectral irradiance outside the atmosphere and therefore shows the highest magnitude (AM0). The direct beam at the surface shows attenuation due to atmospheric absorption and scattering (approximately AM1 under mid-latitude conditions), matching the SMARTS output. The total horizontal irradiance represents the hemispherical irradiance received by a horizontal surface and includes the projected direct component $E_{\mathrm{dir,n}} \cos \theta_S$ and the diffuse sky contribution $E_{\mathrm{dif,h}}$.

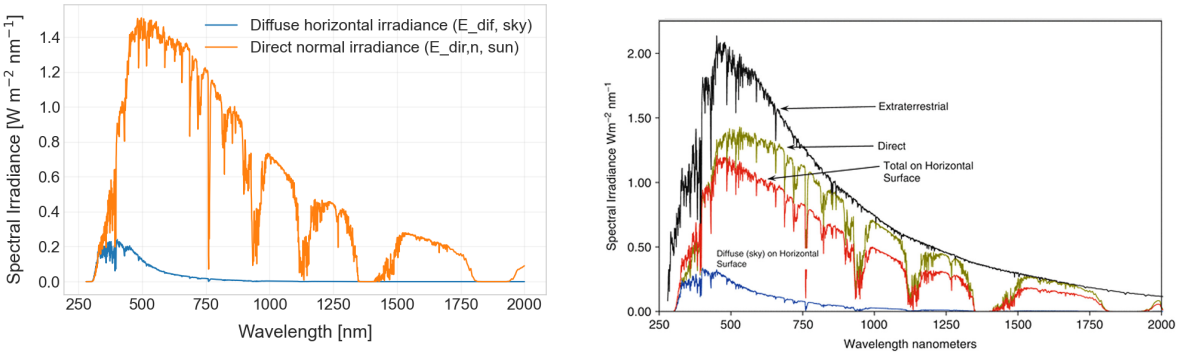


Figure 6.28: Comparison between the SMARTS-generated spectra (left) and representative literature spectra (right). The extraterrestrial curve (black, literature) represents the solar spectrum outside the atmosphere. The direct curve (green, literature) corresponds to the attenuated beam at the surface, matching the SMARTS output. The total horizontal curve (red, literature) includes both direct and diffuse contributions. The blue curve shows diffuse sky irradiance.

Despite the unknown acquisition time and location of the literature example, the spectral magnitudes and characteristic absorption structures show agreement. The absorption band around approximately $760\,\mathrm{nm}$ corresponds to oxygen O_2 absorption. Absorption features near $940\,\mathrm{nm}$, $1130\,\mathrm{nm}$, and beyond $1380\,\mathrm{nm}$ are associated with water vapour H_2O [175]. These atomic and molecular absorption signatures appear at identical wavelength regions in both the literature reference and the SMARTS-generated spectrum, which indicate representative atmospheric behaviour.

The diffuse component generated by SMARTS shows the expected smoother spectral behaviour and lower magnitude compared to the direct beam, consistent with Rayleigh scattering by air molecules and Mie scattering by aerosols [174]. In wavelength regions where strong gaseous absorption suppresses the direct beam, around 250 - 400 nm, the relative contribution of diffuse irradiance increases, matching the physical expectation.

In addition to qualitative comparison, the spectral energy distribution over wavelength regions is evaluated. According to [176], approximately 5% of total solar energy lies in the ultraviolet (300-400 nm), 43% in the visible (400-700 nm), and 52% in the near-infrared (700-2500 nm), as illustrated in Figure 6.29.

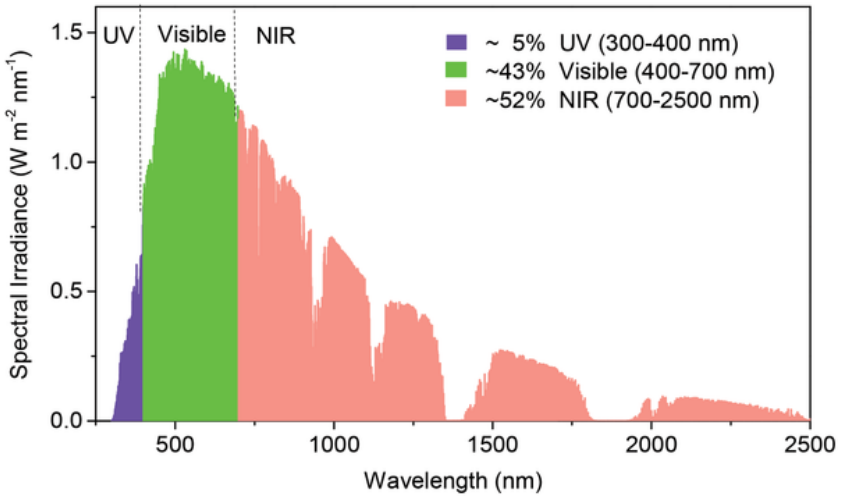


Figure 6.29: Expected percentage distribution of total solar irradiance across spectral regions (UV, visible, and near-infrared) [176]. Approximately 5% lies in the UV (300-400 nm), 43% in the visible (400-700 nm), and 52% in the near-infrared (700-2500 nm).

Integrating the SMARTS-generated direct normal irradiance over the wavelength range 250–2500 nm gives a distribution of 2.79% in the ultraviolet (300–400 nm), 42.82% in the visible (400–700 nm), and 54.38% in the near-infrared (700–2500 nm).

The visible and near-infrared fractions closely match the literature values, while the ultraviolet fraction is slightly lower. This deviation is not significant and can be attributed to differences in location, atmospheric conditions, or solar zenith angle between the SMARTS simulation and the reference example.

The curve alignment in visible and near-infrared proportions, together with the correct representation of atmospheric absorption bands and physically consistent relative magnitudes between direct and diffuse components, indicates that the SMARTS-generated illumination is representative for surface radiative conditions.

While the internal numerical formulation of SMARTS is not evaluated in this work, the agreement in magnitude, spectral structure, absorption features, and energy distribution supports its suitability for the radiometric simulations performed in this thesis.

6.2.4. Unit Testing

To validate if the radiation was correctly converted to reflectance, unit testing is performed using the equations from subsection 5.3.3.

TOA reflectance is defined in Equation 6.12.

$$\rho_{\text{TOA,band}} = \frac{\pi \, \bar{L}_{\lambda,\text{band}} \, d^2}{\bar{E}_{\text{band}} \cos\theta_S}. \tag{6.12}$$

Here, π, $\cos\theta_S$, and the Earth-Sun distance d, expressed in AU, are unitless. The dimensional consistency therefore depends on $\bar{L}_{\lambda,\text{band}}$ and $\bar{E}_{\text{band}}$.

The band-averaged solar irradiance is again given in Equation 6.13 [139].

$$\bar{E}_{\text{band}} = \frac{\int E_{\odot}(\lambda)\, R(\lambda)\, d\lambda}{\int R(\lambda)\, d\lambda}. \tag{6.13}$$

Substituting units gives

$$\bar{E}_{\text{band}} = \frac{\int \left[\text{W m}^{-2}\,\text{nm}^{-1}\right] \cdot [-] \cdot [\text{nm}]}{\int [-] \cdot [\text{nm}]} = \left[\text{W m}^{-2}\,\text{nm}^{-1}\right]. \tag{6.14}$$

Here, $E_{\odot}(\lambda)$ has units $\text{W m}^{-2}\,\text{nm}^{-1}$, the spectral response $R(\lambda)$ is unitless, and $d\lambda$ carries units of nm. The numerator and denominator contain both [-] [nm] which cancel out, and the equation restores spectral irradiance units.

The band-integrated radiance and band-averaged spectral radiance is given in Equation 6.15.

$$L_{\text{band}} = \int L(\lambda)\, R(\lambda)\, d\lambda = \int \left[\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}\right] \cdot [-] \cdot [\text{nm}] = \left[\text{W m}^{-2}\,\text{sr}^{-1}\right], \tag{6.15}$$

$$\bar{L}_{\lambda,\text{band}} = \frac{L_{\text{band}}}{\Delta\lambda} = \frac{\left[\text{W m}^{-2}\,\text{sr}^{-1}\right]}{[\text{nm}]} = \left[\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}\right]. \tag{6.16}$$

Here, $L(\lambda)$ has units $\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}$. Integration over wavelength removes the spectral term, and division by $\Delta\lambda$ restores it.

Substituting Equation 6.14 and Equation 6.16 into Equation 6.12 gives

$$\rho_{\text{TOA,band}} = \frac{[-] \cdot \left[\text{W m}^{-2}\,\text{sr}^{-1}\,\text{nm}^{-1}\right] \cdot [-]^2}{\left[\text{W m}^{-2}\,\text{nm}^{-1}\right] \cdot [-]} = \left[\text{sr}^{-1}\right]. \tag{6.17}$$

Since the steradian is dimensionless, $\left[\text{sr}^{-1}\right] = [-]$.

Therefore, Equation 6.12 gives a unitless quantity, confirming that the implemented radiance-to-reflectance conversion is dimensionally consistent and physically correct.

6.2.5. Daylight Cycle

To verify the propagation of the sun visually throughout the day, the same location is plotted from sunrise to sunset, keeping the observer (satellite) and target at fixed positions to isolate temporal illumination effects.

For this experiment, the target is fixed at latitude $53.0°$ N and longitude $0.0°$ E, at sea level ($h = 0$ m). The observer (satellite) is fixed at latitude $58.0°$ N and longitude $0.0°$ E, with altitude $h = 617$ km, targeting approximately 40 degrees off-nadir. The date is set to 21 June 2025, and time is propagated from 04:00 to 20:00 in steps of 15 minutes. All other parameters, including wind speed and DEM seed, are kept constant, so that the observed variation is driven by the changing solar geometry only.

Figure 6.30 shows the result. Visual inspection shows that near sunrise, around 04:00–05:00, and sunset, around 19:30–20:15, sun glints are strong and the water shows a red tone. This occurs because, at low solar elevation angles, the optical path length through the atmosphere increases approximately as

$1/\cos\theta_s$. Rayleigh scattering scales with wavelength as λ^{-4}, meaning shorter wavelengths (blue) are scattered significantly stronger than longer wavelengths (red), and are consequently more diminished in the atmosphere. Therefore, the reflected radiance shifts toward the red band during sunrise and sunset.

As the Sun rises, the solar elevation increases and the atmospheric path length decreases. The colour tone of the water becomes more neutral and remains approximately constant during the central part of the day. Radiance gradually increases until approximately 12:00, when the Sun reaches its highest elevation and the incident irradiance on the surface is maximal. After local noon, the radiance decreases as solar elevation reduces toward sunset.

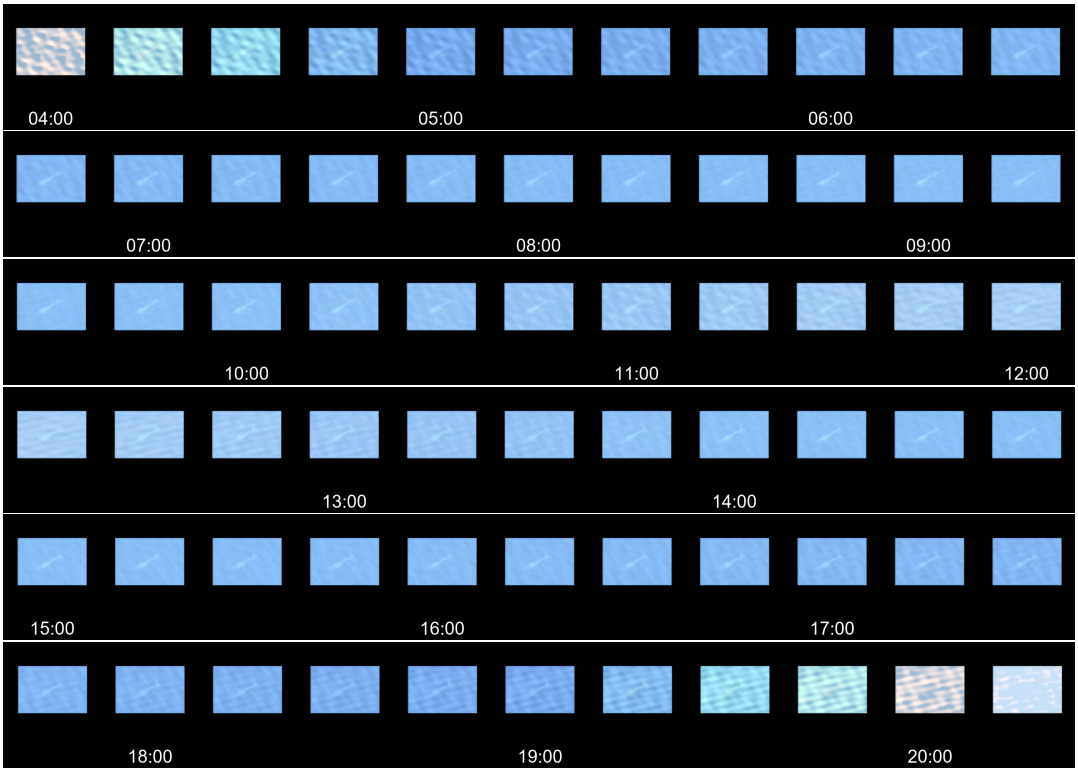


Figure 6.30: Rendered sequence from 04:00 to 20:00 showing the daylight cycle at fixed observer and target positions. Radiometric variations are solely caused by changing solar geometry. *Satellite image © 2022 Maxar Technologies.*

Figure 6.31 shows the temporal evolution of the band-integrated radiance magnitude at the sensor. The upper plot presents the spatial mean and the 95th percentile of the radiance vector norm $\|\mathbf{L}\|$ over the masked DEM footprint, while the bottom plot shows the mean radiance per RGB band.

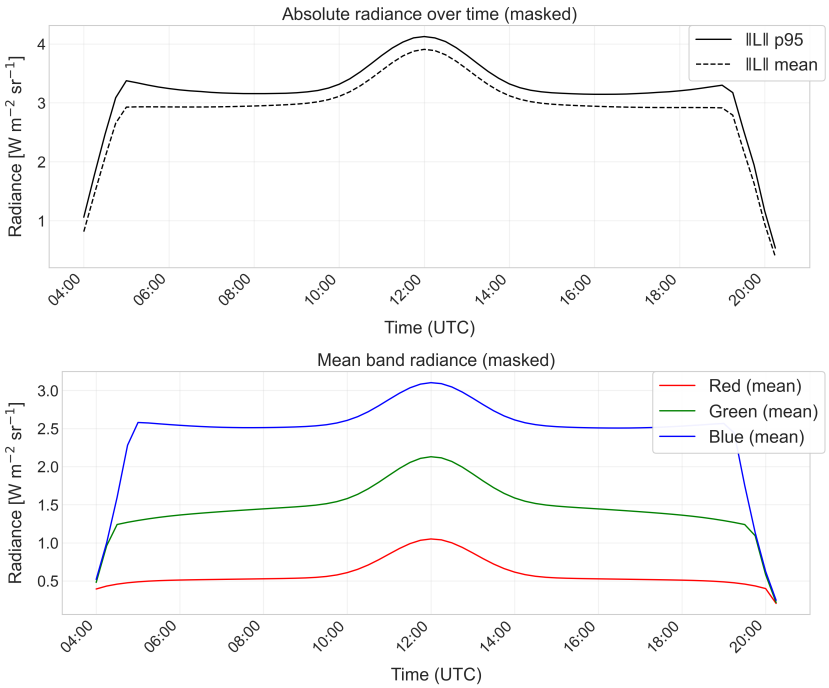


Figure 6.31: Top: Absolute radiance magnitude (mean and 95th percentile) over time for the fixed-observer configuration. Bottom: Mean RGB-band radiance over time. The curves follow the expected cosine-driven daylight cycle, with a midday maximum and lower values toward sunrise and sunset, modulated by weak specular contributions.

The solar irradiance received at the top of the atmosphere is given by the solar constant, $E_0 \approx 1361\,\mathrm{W\,m^{-2}}$, which is projected onto the surface according to the cosine of the solar zenith angle, $E = E_0 \cos\theta_s$, where θ_s is the solar zenith angle. Figure 6.32 shows the cosine behaviour over time.

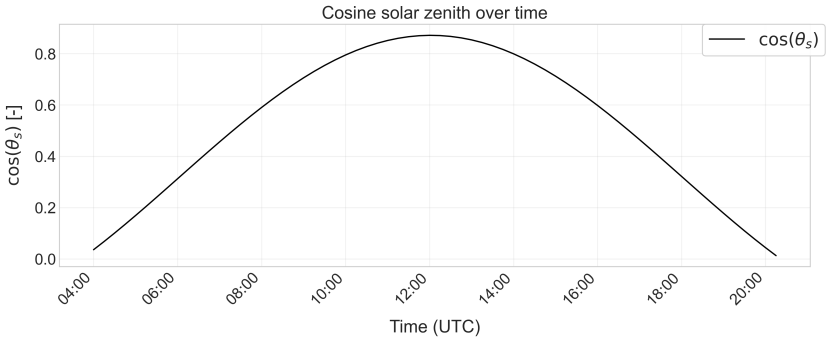


Figure 6.32: Temporal evolution of the cosine of the solar zenith angle, $\cos(\theta_s)$, for the fixed observer-target geometry. The curve follows the expected daylight-cycle variation that governs the projected direct solar irradiance.

The overall magnitude of the radiance plots between 10:00 and 14:00 follow the expected cosine-driven behaviour. Starting at 04:00, radiance increases as the solar zenith angle decreases and $\cos\theta_s$ increases. A maximum is reached around 12:00, when the Sun attains its highest elevation. After local noon, the radiance decreases toward sunset, as expected with the reduction of $\cos\theta_s$.

Between 06:00 and 10:00, and 14:00 and 20:00, the magnitude of the curve is approximately constant. This indicates diffuse-driven behaviour, while the specular peaks are caused by the microfacet BSDF around noon. During the dark hours between 20:00 and 04:00, the magnitude drops because the Sun falls below the effective illumination geometry and the direct component vanishes. The temporal propagation is free of discontinuities, confirming consistency of the implemented illumination model.

The mean radiance magnitude reaches values in the order of $3.8\,\mathrm{W\,m^{-2}\,sr^{-1}}$ around noon. These values

are band-integrated at-sensor radiances as returned directly by Mitsuba. They are therefore not directly comparable to the solar constant $E_0 \approx 1361\ \mathrm{W\,m^{-2}}$, which represents broadband hemispherical irradiance at the top of the atmosphere.

To verify that the magnitude is physically plausible, the band-integrated irradiance can be considered. For a given spectral band,

$$E_{\text{band}} = \int E_\lambda(\lambda) R_{\text{band}}(\lambda)\, d\lambda,$$

which typically gives values in the order of $100{-}200\ \mathrm{W\,m^{-2}}$ in the visible range [177]. Projected onto the surface, this becomes $E_{\text{band}} \cos\theta_s$.

Assuming Lambertian behaviour, the reflected band-integrated radiance is given by Equation 6.18.

$$L_{\text{expected}} = \frac{\rho}{\pi} E_{\text{band}} \cos\theta_s \tag{6.18}$$

For ocean water, its expected reflectance lies in the range $\rho \approx 0.03{-}0.1$ [178]. Taking a case with $\rho = 0.06$, $\cos\theta_s \approx 0.9$, following the maximum in Figure 6.32, and $E_{\text{band}} = 200\ \mathrm{W\,m^{-2}}$, gives a peak estimation of

$$L \approx \frac{0.06}{\pi} \cdot 200 \cdot 0.9 \approx 3.44\ \mathrm{W\,m^{-2}\,sr^{-1}}.$$

which is consistent with the simulated peak magnitudes. With that, the observed values therefore fall within the physically expected range.

The per-band radiance curves further show that the blue band shows the highest mean radiance during most of the day, followed by green and red. This ordering is expected, following the shape of the solar spectrum with higher spectral irradiance at shorter wavelengths, and the higher relative atmospheric transmission of blue at high solar elevation.

With that, the magnitude, spectral ordering, and temporal evolution of the radiance curves confirm that the daylight cycle and radiometric scaling are physically consistent within the implemented rendering pipeline.

6.2.6. Maximum Glint

A similar experiment as in subsection 6.2.5 was repeated, but in this case, while propagating the Sun over time, the observer (satellite) was repositioned at each time step to the theoretical specular reflection direction corresponding to maximal sun glint. The objective is to confirm that the implemented microfacet BSDF produces the expected maximum radiometric response under perfect specular alignment.

The resulting rendered sequence is shown in Figure 6.33. In contrast to the fixed-satellite daylight cycle, the specular highlight remains centered in the field of view throughout the day, since the satellite continuously tracks the mirror direction. Strong, visually saturated glint patterns are observed for all daylight hours, with the brightest response occurring around sunrise and sunset.

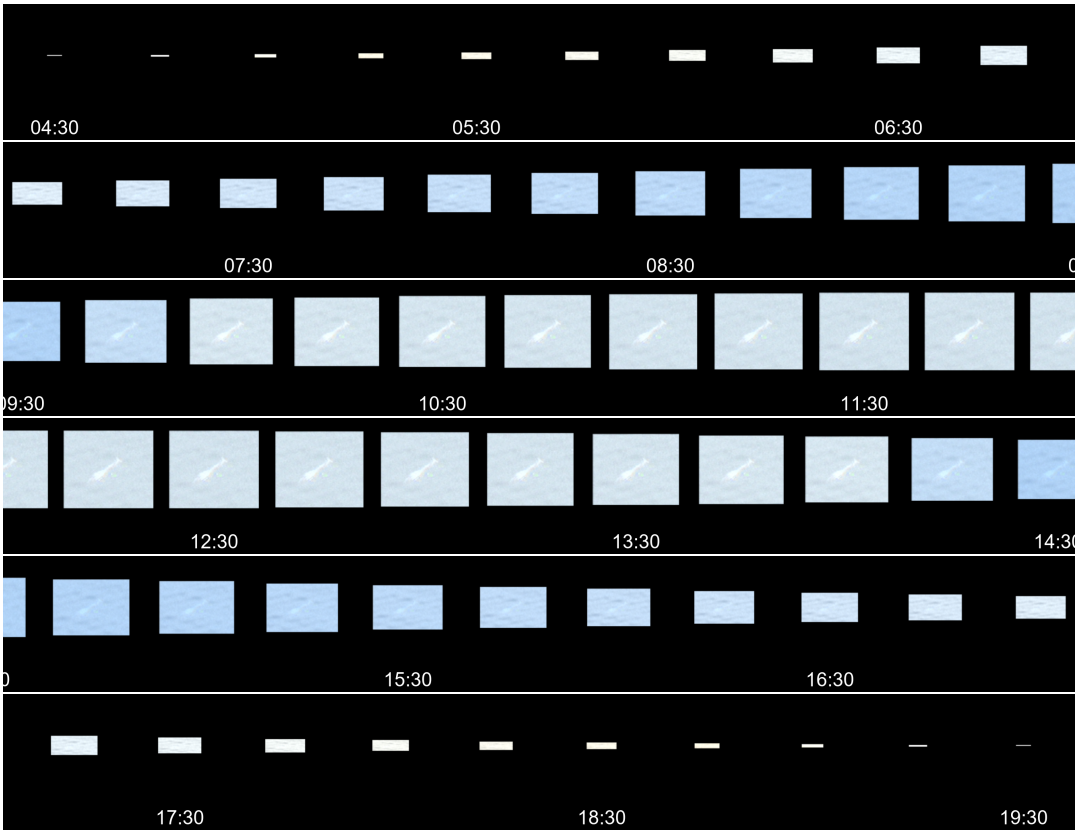


Figure 6.33: Rendered sequence from 04:00 to 19:00 showing the maximum glint cycle at fixed target position. The satellite is dynamically repositioned to satisfy the specular reflection condition at each timestep. *Satellite image © 2022 Maxar Technologies.*

First, the placement of the satellite at maximum glint position was verified geometrically. Figure 6.34 shows three representative configurations at different solar positions during the day. In each case, the Sun vector and viewing direction satisfy the specular reflection law, meaning that the angle of incidence equals the angle of reflection with respect to the local normal. Visually, the satellite lies on the reflected ray of the solar direction, confirming that the geometric setup is correct.

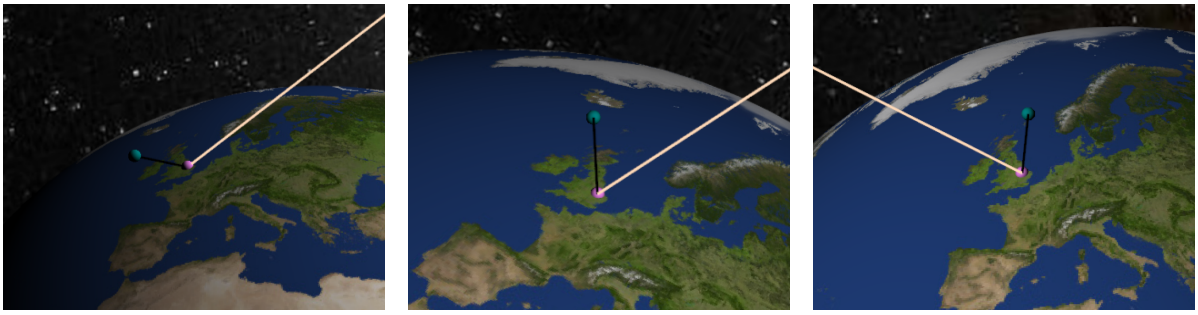


Figure 6.34: Geometric verification of maximum glint positioning at three different times of day. For each configuration, the satellite, in blue, is placed along the specular reflection direction of the incident solar ray with respect to the local surface normal, satisfying the mirror-reflection condition.

The temporal evolution of the radiance magnitude for this configuration is shown in Figure 6.35. The upper plot presents the mean and 95th percentile of the radiance norm, while the lower plot shows the mean RGB band radiance.

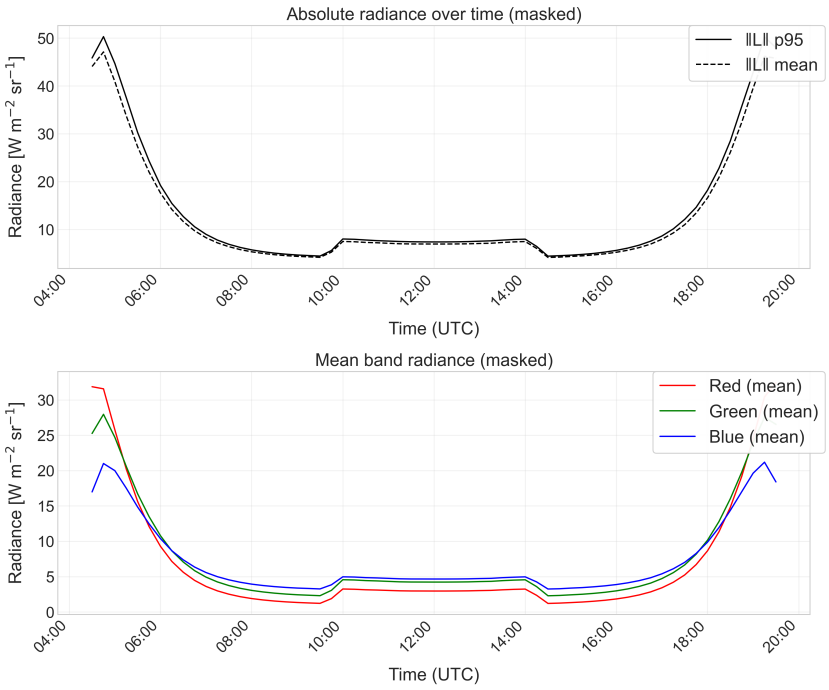


Figure 6.35: Top: Absolute radiance magnitude (mean and 95th percentile) over time for the maximum-glint configuration. Bottom: Mean radiance in the red, green, and blue bands over time. The peaks near sunrise and sunset reflect the strong increase in specular Fresnel reflection under grazing-angle illumination.

This temporal evolution shows a different behaviour compared to the fixed-satellite configuration. The mean radiance magnitude is maximal near sunrise and sunset, approaching $50\ \mathrm{W\,m^{-2}\,sr^{-1}}$. From sunrise until approximately 06:00, the curve decreases very steeply. Between 06:00 and about 09:30 the decrease continues more gradually. Around 10:00 a small increase occurs, after which the curve remains nearly flat between 10:00 and 14:00. After 14:00, the behaviour mirrors the morning pattern: first a slight decrease, then a steeper increase toward sunset.

This behaviour follows directly from the microfacet BRDF formulation. The specular component can be written as

$$L \;\propto\; E_{\mathrm{band}} \cos\theta_s \;\frac{D(\theta_h)\,F(\theta_i)\,G}{4\cos\theta_i \cos\theta_o},$$

where $E_{\mathrm{band}} \cos\theta_s$ is the projected irradiance term, $D(\theta_h)$ is the Beckmann normal distribution function (see Equation 3.5), and $F(\theta_i)$ is the Fresnel term.

Near sunrise and sunset, the solar zenith angle θ_s is large, so $\cos\theta_s$ is small and the projected irradiance term is reduced. However, the incidence angle θ_i approaches grazing conditions. In this regime, the Fresnel term $F(\theta_i)$ increases strongly as $\theta_i \to 90°$. This grazing-angle amplification dominates over the reduction in $\cos\theta_s$, producing the high radiance peaks at the edges of the day.

As the Sun rises, θ_i decreases and the Fresnel term drops rapidly. This explains the steep decline between 04:30 and 06:00. During the central part of the day, $\cos\theta_s$ is maximal, but the Fresnel term is much smaller and varies only slowly with angle. Since $D(\theta_h)$ remains near its peak under enforced specular alignment, the product of all terms changes only weakly, leading to the observed flattening between 10:00 and 14:00.

The RGB curves follow the same structure. At low solar elevation, increased atmospheric path length suppresses shorter wavelengths more strongly, so the red band dominates. Around noon, the atmospheric path length decreases the blue band slightly overpasses green and red.

To verify whether the maximum radiance peaks of approximately $50\ \mathrm{W\,m^{-2}\,sr^{-1}}$ are physically achievable, the same Lambertian reference formulation as in subsection 6.2.5 is used.

$$L_{\text{expected}} = \frac{\rho}{\pi} E_{\text{band}} \cos\theta_s. \tag{6.19}$$

Under strong sunglint conditions, effective water reflectance can become very large due to specular Fresnel reflection. Measurements and modeling studies of ocean sunglint report reflectance values exceeding $\rho \approx 5$ for large solar zenith angles and near-specular viewing geometries [179].

Taking a sunglint reflectance of $\rho = 6.0$, a band-integrated irradiance of $E_{\text{band}} = 200\ \text{W m}^{-2}$ in the visible range, consistent with the values used in subsection 6.2.5, and a low-sun configuration with $\cos\theta_s = 0.1$, as occurring in the borders of Figure 6.32, Equation 6.19 gives

$$L_{\text{expected}} = \frac{6}{\pi} \cdot 200 \cdot 0.1 \approx 38.2\ \text{W m}^{-2}\,\text{sr}^{-1}.$$

For slightly larger effective reflectance $\rho \approx 8$, as reported at lower surface-level wind speeds [179], the resulting radiance increases to

$$L \approx \frac{8}{\pi} \cdot 200 \cdot 0.1 \approx 50.9\ \text{W m}^{-2}\,\text{sr}^{-1},$$

which brackets the simulated peak magnitude of $\sim 50\ \text{W m}^{-2}\,\text{sr}^{-1}$.

Therefore, the rendered maximum-glint radiance peaks fall within the physically expected range for strong specular ocean reflection.

6.2.7. Satellite Data Comparison

To validate the end-to-end rendering pipeline, the generated images are compared against real satellite observations from the ESA Φ-Sat-2 mission. Φ-Sat-2 is a 6U CubeSat developed within the ESA Φ-lab programme to test novel onboard AI algorithms for Earth observation. The satellite operates in a Sun-synchronous low Earth orbit at an altitude of approximately 510 km [180], comparable to the benchmark configuration.

Ideally, the validation would be performed using L0 or L1A/B satellite products, comparing radiance values directly to the radiance output generated by the Mitsuba renderer. In principle, L0 Φ-Sat-2 data could be converted to L1 radiance using the radiometric calibration coefficients typically applied through the relation given in Equation 6.20 [139]. Such calibration ensures that the recorded digital numbers are converted into physically meaningful radiance values and is normally applied as part of the standard satellite processing chain.

$$L = \text{GAIN} \cdot \text{DN} \cdot \left(\frac{\text{abscalfactor}}{\text{effectivebandwidth}} \right) + \text{OFFSET} \tag{6.20}$$

In Equation 6.20, L denotes the spectral radiance, DN represents the recorded digital number, GAIN and OFFSET are radiometric calibration coefficients, while the absolute calibration factor and effective bandwidth account for the spectral characteristics of the sensor.

Although both L0 and L1 products are available for Φ-Sat-2, the required radiometric calibration coefficients are not publicly provided. As a result, the L0 data cannot be converted to radiance. Conversely, the Mitsuba rendering pipeline outputs radiance values that correspond to an L1 representation and cannot be converted back to L0 data without introducing arbitrary calibration factors.

Alternatively, the L1C reflectance products could in principle be converted back to radiance by reversing the reflectance conversion described in subsection 5.3.3. However, this would require knowledge of the satellite spectral response curves. These response functions are not publicly available for Φ-Sat-2, and the spectral response curves used in the rendering pipeline are themselves approximated. Consequently, a radiance-level comparison would largely be dominated by the uncertainty in these unknown spectral response functions rather than by the physical validity of the rendering model.

For these reasons, the validation is performed in the reflectance domain. It is assumed that the spectral response curves approximated within the rendering pipeline are patternwise representative of Φ-Sat-2, allowing reflectance values to be compared directly between the simulated and observed data.

Therefore, Φ-Sat-2 L1C data are used for the comparison. The L1C product corresponds to Top-of-Atmosphere reflectance, whereas the L1A and L1B products represent radiance values [181]. The stored reflectance values are encoded with a scaling factor of 10 000 and are converted to physical reflectance by dividing by this factor [182]. Validation is performed using the following two Φ-Sat-2 images [182]:

- PHISAT-2_L1_000002103_20250423144634_20250423144637_B69C2A81
- PHISAT-2_L1_000001987_20250410143947_20250410143950_B05E6C3E

For clarity, these datasets will be abbreviated by their final identifiers as A81 and C3E respectively. Both images correspond to the same geographic region but were acquired under different viewing geometries: A81 at an off-nadir angle of 11.3° and C3E at 8.2°. The maximum sun-glint angle for this scene is approximately 20.4°. The images are shown in Figure 6.36. For visualization purposes, the reflectance values are multiplied by the storage factor of 10 000 and gamma corrected.

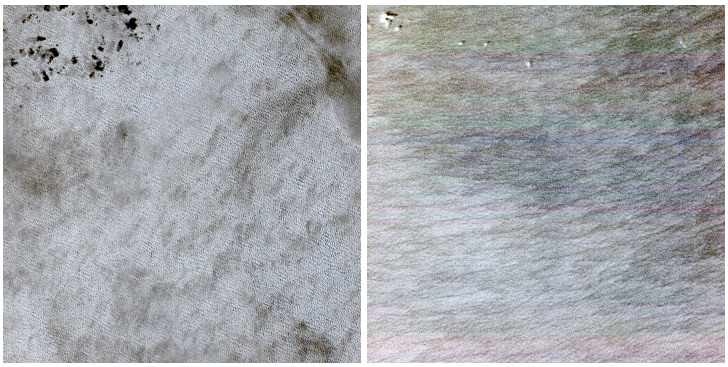


Figure 6.36: The L1C input images of Φ-Sat-2, representing Top of Atmosphere reflectance data. On the left: A81, on the right: C3E. For visualization only, both images are loaded and bit-scaled by factor 10000 and gamma corrected. [182]

Rendering an off-nadir image of size 4096×4096, matching the native dimensions of the satellite product, would require substantial computational resources when performed repeatedly for parameter sweeps. Therefore, a central crop of 512×512 pixels is extracted from the satellite image for the comparison. This corresponds to $262\,144$ pixels, from which reflectance statistics are computed. The resulting statistics for both datasets are listed in Table 6.5.

Table 6.5: Channel Statistics for A81 and C3E Φ-Sat-2 images [182].

Dataset	Channel	N Samples	Min	Max	Mean	p1	p50	p99
A81	R	262144	0.0353	0.3603	0.1410	0.0684	0.1385	0.2398
A81	G	262144	0.0483	0.3250	0.1571	0.0887	0.1552	0.2441
A81	B	262144	0.0885	0.3427	0.1877	0.1298	0.1865	0.2558
C3E	R	262144	0.0123	0.0362	0.0194	0.0152	0.0192	0.0256
C3E	G	262144	0.0264	0.0457	0.0330	0.0291	0.0329	0.0382
C3E	B	262144	0.0574	0.0726	0.0634	0.0601	0.0633	0.0674

The objective is then to generate a geometrically equivalent ocean scene using the rendering pipeline, where the satellite position, target position, and acquisition time match those of the Φ-Sat-2 observation. The rendered output is expected to exhibit reflectance values of comparable magnitude.

To evaluate the influence of the underlying texture, five ocean surface patches from the Whales from Space dataset are used as input textures. These patches originate from different geographic regions and sensors and exhibit varying radiometric characteristics, as shown in Figure 6.37.

Figure 6.37: Input patches for the rendering pipeline, build from only cropped Whales from Space dataset [100], chosen with variation in radiometric representation. From left to right: Auckland 2006, Auckland 2011, Maui 2015, Pelagos 2016, Valdes 2016. *Satellite image © 2022 Maxar Technologies.*

Since the Φ-Sat-2 satellite position is not explicitly provided in the product metadata, it is reconstructed using the acquisition timestamp and orbital elements. Two TLE files surrounding the acquisition time are retrieved and propagated using the Python library tools_tle [183]. From these orbital elements and the acquisition time, the satellite latitude and longitude are computed and used as inputs to the rendering pipeline.

The target position corresponds to the location of the observed scene. This information is retrieved from the geolocation file containing the latitude and longitude mapping for each pixel. As the rendering algorithm assumes that the sensor boresight points directly towards the target, the scene centre is extracted and used as the target location.

The satellite position, target position, acquisition time, and an ocean texture patch are then provided to the rendering pipeline to generate a reflectance image. From the resulting image, statistical measures such as the mean, p1, p50, and p99 reflectance are computed. The comparison focuses on numerical reflectance statistics rather than visual similarity, since the rendering pipeline intentionally uses the original input textures, similar to what is used for generating the dataset.

It was observed that the simulated reflectance values are highly sensitive to the surface roughness parameter, which is determined by the ocean surface wind speed. Because the wind speed at the time of the satellite acquisition is unknown, a parameter sweep between 2.0 and 12.0 m/s is performed. This range corresponds to the wind conditions used during dataset generation.

For each wind speed, the surface roughness is computed using the Cox-Munk model: $\sigma^2 = 0.003 + 0.00512 \cdot \text{wind\_speed}$, with $\alpha = \sqrt{\sigma^2}$, as described in section 5.3 [138]. The resulting roughness parameter is then used by Mitsuba to simulate the ocean surface BRDF.

By comparing the reflectance statistics of the rendered images with those of the Φ-Sat-2 observations, an approximate wind speed that reproduces the observed reflectance distribution can be estimated. An example result for the Auckland 2006 texture is shown in Figure 6.38, while the remaining locations and numerical results for all RGB channels are provided in Appendix F.

The dots represent the measured reflectance values of the Φ-Sat-2 image at the estimated representative wind speed, while the curves show the reflectance values generated by the renderer for the same satellite, target, and solar geometry across different wind speeds.

Notably, the resulting reflectance curves show substantial overlap with the original for both imaging conditions. For C3E specifically, the measured reflectance values lie near the lower bound of the explored wind-speed range. This may indicate that the actual spectral response of Φ-Sat-2 differs from the WorldView-2-based approximation used in the rendering pipeline. If the true Φ-Sat-2 spectral response curves are broader or have a higher effective response over the relevant wavelength range, more radiance would be integrated by the sensor, which would result in slightly higher reflectance values than those produced by the renderer.

Even so, the wind-speed range was not extended below 2.0 m/s to force a closer match. Such low surface wind speeds are less likely for the considered conditions and are also not included in the dataset generation procedure [145]. For that reason, keeping the same physically plausible range was considered more appropriate than optimizing the agreement through extrapolation.

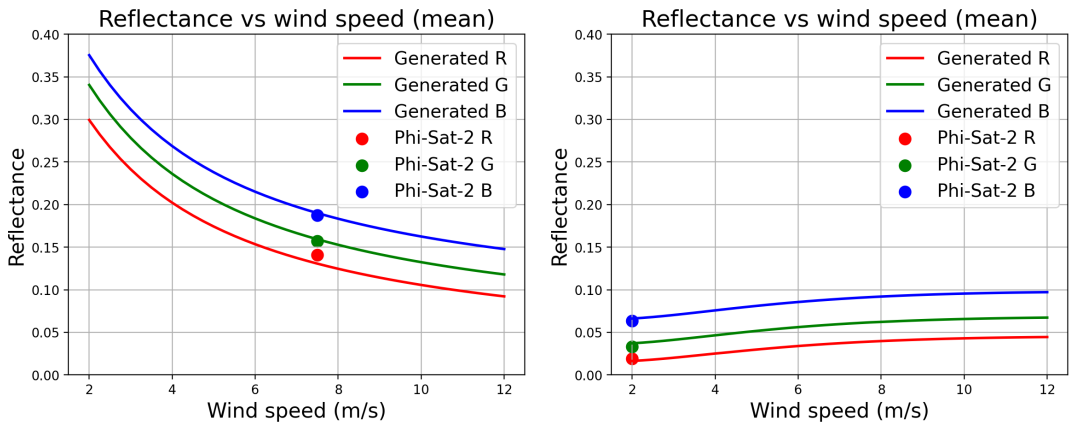


Figure 6.38: Mean reflectance as a function of wind speed for the Auckland 2006 input texture for the A81 (left) and C3E (right) observation geometries. The curves represent the reflectance values for the R, G, and B spectral bands of the generated images, while the markers indicate the true Φ-Sat-2 reflectance values. The opposite trends indicate that the renderer consistently reproduces the changing balance between specular and diffuse reflection under different viewing geometries.

It is further observed that the two datasets show opposite trends with increasing wind speed: the reflectance curves for C3E rise slightly, whereas those for A81 decrease. This difference can be explained by the observation geometry, visualized in Figure 6.39. For A81, acquired at 11.3° off-nadir, the viewing direction lies closer to the maximum sunglint condition, given a maximum glint angle of 20.4°. In this configuration, the microfacet-based sun-glint component contributes more strongly to the total reflectance, which is consistent with the substantially larger reflectance magnitude of A81 compared to C3E, approximately a factor four. As with increasing wind speed the surface becomes rougher, the specular reflection is distributed over a wider range of directions, reducing the radiance concentrated towards the sensor and therefore decreasing the observed reflectance.

For C3E, acquired at 8.2° off-nadir, the sensor is positioned further away from the dominant glint configuration. In that case, the diffuse and broader non-specular contributions become relatively more important. A moderate increase in surface roughness can then redirect more reflected energy towards the sensor, producing the slight increase in reflectance observed in the plots. The opposite trends of A81 and C3E therefore indicate that the renderer responds consistently to changes in the balance between specular and diffuse reflection under different viewing geometries.

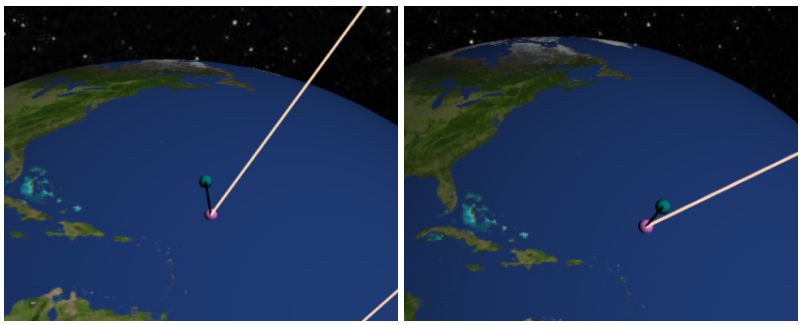


Figure 6.39: Observation geometry of the two Φ-Sat-2 acquisitions used for validation. Left: A81 scene acquired at 11.3° off-nadir, closer to the maximum sunglint configuration of 20.4°. Right: C3E scene acquired at 8.2° off-nadir, positioned further away from the dominant glint direction.

Lastly, the variance between different input textures was assessed. For each dataset, the reflectance curves of all five input images were grouped per spectral channel, as shown in Figure 6.40 and Figure 6.41. It is observed that, for the same scene geometry and wind speed, the rendered reflectance remains consistent across the different input textures per band.

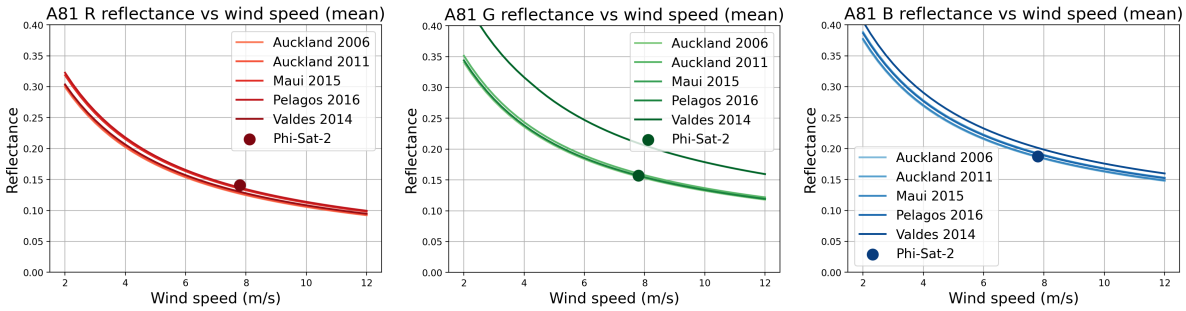


Figure 6.40: Mean reflectance as a function of wind speed for the A81 observation geometry for R, G, B bands of five ocean input textures. The curves represent generated reflectance values for the R, G, and B spectral bands, while the markers indicate the true reflectance of the Φ-Sat-2 observation for each band.

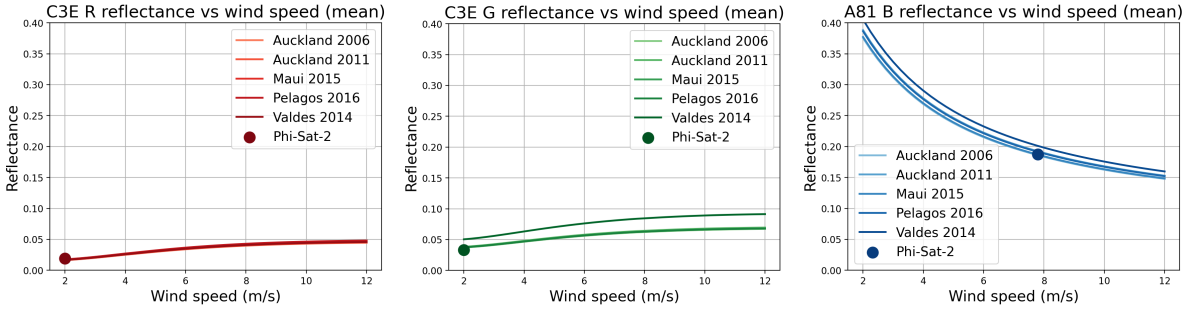


Figure 6.41: Mean reflectance as a function of wind speed for the C3E observation geometry for five ocean input textures. The curves represent generated reflectance values for the R, G, and B spectral bands, while the markers indicate the true reflectance of the Φ-Sat-2 observation for each band.

Still, one exception occurs in the green band for the Valdes input texture. This deviation is caused by the relatively low original intensity of the input patch, with a mean pixel value of 0.154 compared to the overall mean of 0.513, as shown in Table 6.6.

Table 6.6: Mean reflectance values of the red (R), green (G), and blue (B) bands for the five ocean textures used as input in the rendering pipeline.

Location	R Mean	G Mean	B Mean
Auckland 2006	0.354	0.427	0.452
Auckland 2011	0.526	0.631	0.711
Maui 2015	0.497	0.491	0.471
Pelagos 2016	0.858	0.861	0.860
Valdes 2016	0.076	0.154	0.432
Mean	0.462	0.513	0.585

This affects the internal reflectance matching step of the rendering pipeline under the microfacet-based rescaling procedure described in section 5.3, resulting in differently scaled output values. An improvement would be to normalize the input textures before estimating their radiometric representation, thereby reducing the sensitivity of the texture-dependent scaling to the original image intensity. Despite the omission of input patch normalization beforehand, this effect is not observed for the other locations, which all have mean values close to or above 0.5.

Although the original dataset also shows mismatched output reflectances for Valdes 2016, visualized in Appendix D, these samples were retained during training to reduce the risk of overfitting and to improve generalization to unexpected imaging conditions.

The numerical results, obtained by averaging the reflectance statistics over the renders from all five input locations at the matched wind speed, are given in Table 6.7 and Table 6.8. The expalnded results for all five image renders individually are provided in Appendix F.

The results indicate that the median reflectance values ($p50$) agree closely with the Φ-Sat-2 reference data across all spectral bands, while the variance between different texture inputs remains limited. Using the median $p50$ as the primary comparison metric, the deviation between the generated images and the Φ-Sat-2 observations ranges from approximately 1% to 14%, depending on the spectral band and dataset.

In contrast, the distribution extremes show larger deviations. The lower tail $p1$ differs by approximately 6% to 39%, while the upper tail $p99$ differs by approximately 2% to 73%, depending on the spectral band and dataset. This behaviour is expected, because the distribution extremes are dominated by localized texture artefacts and high-frequency surface structure introduced by the simulated wave field, whereas the median reflectance is more representative of the overall water-surface response.

A comparison between the spectral bands further shows that the green band exhibits the largest deviation for the C3E dataset, with a median difference of approximately 14.2%, compared to 11.1% for the red band and 6.2% for the blue band. For the A81 dataset, the agreement is stronger, with median differences of approximately 4.6% (R), 1.2% (G), and 1.3% (B).

The larger discrepancy in the green band can be explained by several physical factors. First, the green wavelength range is more sensitive to water-leaving radiance contributions influenced by chlorophyll absorption and scattering [184]. Second, this spectral region is affected more strongly by variations in subsurface optical properties and biological constituents [185], which are not explicitly modelled in the simplified water-surface representation used here. Third, residual mismatch between the approximated spectral response curves of the rendering pipeline and the true sensor response of Φ-Sat-2 can further contribute to this difference.

Table 6.7: Reflectance statistics comparing generated images and Φ-Sat-2 A81 reference data for the R, G, and B bands at wind speed 7.8 m/s.

Band	Source	n	Min	Max	Mean	p1	p50	p99
R	Mean Generated	245000	0.056004	0.373489	0.131504	0.095404	0.132123	0.192244
R	Mean Φ-Sat-2	262144	0.0353	0.3603	0.141024	0.0684	0.1385	0.2398
R	Absolute difference	17144	0.020704	0.013189	0.009520	0.027004	0.006377	0.047556
R	Percentage difference (%)	6.54	58.65	3.66	6.75	39.48	**4.60**	19.83
G	Mean Generated	245000	0.068462	0.415577	0.167462	0.114436	0.157131	0.272670
G	Mean Φ-Sat-2	262144	0.0483	0.3250	0.157100	0.0887	0.1552	0.2441
G	Absolute difference	17144	0.020162	0.090577	0.010361	0.025736	0.001931	0.028570
G	Percentage difference (%)	6.54	41.74	27.87	6.60	29.02	**1.24**	11.70
B	Mean Generated	245000	0.091138	0.435007	0.191559	0.150151	0.188908	0.250441
B	Mean Φ-Sat-2	262144	0.0885	0.3427	0.187654	0.1298	0.1865	0.2558
B	Absolute difference	17144	0.002638	0.092307	0.003904	0.020351	0.002408	0.005359
B	Percentage difference (%)	6.54	2.98	26.93	2.08	15.68	**1.29**	2.09

Table 6.8: Reflectance statistics comparing generated images and Φ-Sat-2 C3E reference data for the R, G, and B bands at wind speed 2.0 m/s.

Band	Source	n	Min	Max	Mean	p1	p50	p99
R	Mean Generated	253000	0.007194	0.059596	0.017001	0.012336	0.017076	0.024949
R	Mean Φ-Sat-2	262144	0.0123	0.0362	0.019386	0.0152	0.0192	0.0256
R	Absolute difference	9144	0.005106	0.023396	0.002385	0.002864	0.002124	0.000651
R	Percentage difference (%)	3.49	41.51	64.63	12.30	18.84	**11.06**	2.54
G	Mean Generated	253000	0.016149	0.191875	0.040218	0.027302	0.037569	0.065934
G	Mean Φ-Sat-2	262144	0.0264	0.0457	0.033010	0.0291	0.0329	0.0382
G	Absolute difference	9144	0.010251	0.146175	0.007207	0.001798	0.004669	0.027734
G	Percentage difference (%)	3.49	38.83	319.86	21.83	6.18	**14.19**	72.60
B	Mean Generated	253000	0.032297	0.154972	0.068234	0.053530	0.067245	0.089284
B	Mean Φ-Sat-2	262144	0.0574	0.0726	0.063389	0.0601	0.0633	0.0674
B	Absolute difference	9144	0.025103	0.082372	0.004845	0.006570	0.003945	0.021884
B	Percentage difference (%)	3.49	43.73	113.46	7.64	10.93	**6.23**	32.47

The simulated reflectance values follow the magnitude and distribution of the Φ-Sat-2 observations and remain consistent across the different input textures. This indicates that the rendering pipeline produces reflectance values representative of an ocean surface under the same observation geometry.

Combined with the previously verified radiance computation in subsection 6.2.4, the agreement in reflectance indicates that the radiance levels generated by the renderer are of the correct physical order of magnitude. Within the limits of the available satellite metadata and the reflectance-based comparison, this constitutes an end-to-end validation of the rendering pipeline in the reflectance domain.

6.3. Onboard Detection

This section verifies whether the trained detector can be used as a representative Cue-side model in the integrated benchmark. The evaluation focuses on three aspects: the integrity of the train-validation-test split, the standalone detection performance of the final Cue model, and the stability of the selected validation procedure across folds.

6.3.1. Split Integrity

First, it was verified that the pipeline was implemented correctly and that no data leakage between training, validation, and test splits occurred.

To confirm that *Auckland2006* and *Pelagos2016* were completely excluded from the training and validation data and only present in the final test set, the generated COCO annotation files were inspected directly using the following commands:

- `grep -E "Auckland2006|Pelagos2016" trainval_without_holdout_test.json`
  to verify that no images from the test locations are present in the training/validation split. This command should return no output.
- `grep -E "Auckland2011|Ignacio2017|Maui2015|Valdes2012|Valdes2014|Valdes2016|Witsand2009" trainval_without_holdout_test.json`
  to confirm that all remaining locations are included in the training/validation data.
- `grep -E "Auckland2006|Pelagos2016" test_holdout_only.json`
  to verify that the test split contains the designated holdout locations.
- `grep -E "Auckland2011|Ignacio2017|Maui2015|Valdes2012|Valdes2014|Valdes2016|Witsand2009" test_holdout_only.json`
  to ensure that no training locations are present in the test split. This command should also return no output.

Figure 6.42 shows the outputs of these steps, confirming that the final test evaluation is performed on unseen data, thus the reported performance is not affected by data leakage.

```
(deimv2) nduursma@iv-mind:~/DEIMv2-main_1/results/reflection_offnadir_glint_255$ grep -E
"Auckland2006|Pelagos2016" trainval_without_holdout_test.json
(deimv2) nduursma@iv-mind:~/DEIMv2-main_1/results/reflection_offnadir_glint_255$ grep -E
"Auckland2011|Ignacio2017|Maui2015|Valdes2012|Valdes2014|Valdes2016|Witsand2009" trainval
_without_holdout_test.json | head
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000000_F_5deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000001_F_10deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000002_F_15deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000003_F_20deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000004_F_25deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000005_F_30deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000006_F_35deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000007_F_40deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000008_F_45deg.PNG",
      "file_name": "Auckland2011/Auckland_SRW_WV2_PS_20110827_BO_000009_F_50deg.PNG",
(deimv2) nduursma@iv-mind:~/DEIMv2-main_1/results/reflection_offnadir_glint_255$ grep -E
"Auckland2006|Pelagos2016" test_holdout_only.json | head
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000036_F_5deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000037_F_10deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000038_F_15deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000039_F_20deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000040_F_25deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000041_F_30deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000042_F_35deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000043_F_40deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000044_F_45deg.PNG",
      "file_name": "Pelagos2016/PelagosIm2_FW_WV3_PS_20160619_BO_000045_F_50deg.PNG",
(deimv2) nduursma@iv-mind:~/DEIMv2-main_1/results/reflection_offnadir_glint_255$ grep -E
"Auckland2011|Ignacio2017|Maui2015|Valdes2012|Valdes2014|Valdes2016|Witsand2009" test_hol
dout_only.json
(deimv2) nduursma@iv-mind:~/DEIMv2-main_1/results/reflection_offnadir_glint_255$
```

Figure 6.42: Verification of the location-based data split. No occurrences of the holdout locations (*Auckland2006*, *Pelagos2016*) are found in the training/validation annotations, while they are present in the test annotations. Conversely, all training locations appear only in the training/validation split and are absent from the test set, confirming the absence of data leakage.

Similarly, it was verified if there was no leakage from the 5% validation set into the training data. For that, a Python script was written that loads instances_train.json and instances_val.json, and checks for overlap between the file names. Figure 6.43 shows the result and confirms that there is no data leakage between validation and training data.

```
============================== 1 passed in 0.70s ===============================
PASSED [100%]Train images: 11051
Val images: 582
ID overlap: 0
File overlap: 0

Process finished with exit code 0
```

Figure 6.43: Verification of the training–validation split. The number of images in the training and validation sets is reported, together with the overlap in image IDs and file names. Both overlap values are zero, confirming that the 5% validation set is fully disjoint from the training data.

6.3.2. Performance Metrics

Next, the performance of the final onboard detection model is evaluated on the held-out test set. Table 6.9 summarizes the validation and test performance obtained for the final model trained on the reflection off-nadir dataset with radiometric effects.

Table 6.9: Final model validation and test performance on reflection off-nadir data with radiometric effects.

Set	AP_{50}	$AP_{50:95}$	F1	Best F1 Thr
Validation	0.894	0.470	0.962	0.380
Test	0.757	0.343	0.930	0.296

As shown in Table 6.9, performance on the validation set is higher than on the held-out test set across all reported metrics. AP_{50} decreases from 0.894 on validation to 0.757 on test, while $AP_{50:95}$ decreases from 0.470 to 0.343. This gap is expected given the composition of the two splits.

The validation set consists of 5% of the non-test data and is sampled from the same set of locations as the training data. Although the validation images are fully disjoint from the training images, they still contain similar whale patterns and background structure, given they represent the same underlying patch at a different off-nadir angle. As a result, the validation set remains closer to the training distribution than the held-out test set.

By contrast, the test set contains the fixed holdout locations *Auckland2006* and *Pelagos2016*, which are fully excluded from both training and validation, as verified in subsection 6.3.1. The lower test performance therefore reflects the expected generalization gap between in-distribution validation data and unseen operational conditions. At the same time, the test results confirm that the final model retains strong detection performance under fully unseen location conditions, with an AP_{50} of 0.757 and an F1-score of 0.930.

The implications of these quantitative results are examined further through qualitative detection examples in subsection 6.3.3 and through the per-location variance analysis in subsection 6.3.4.

6.3.3. Detection Performance

The qualitative detection performance of the model on the test set is visualized and assessed. The detection confidence threshold is fixed to the value that maximizes the F1-score on the validation set (0.380). While it is not tuned on the test set, this avoids optimistic bias and reflects operational conditions, where no prior knowledge of the test distribution is available.

Detection examples for both Auckland2006 and Pelagos2016 are visualized in Figure 6.44 and Figure 6.45, respectively.

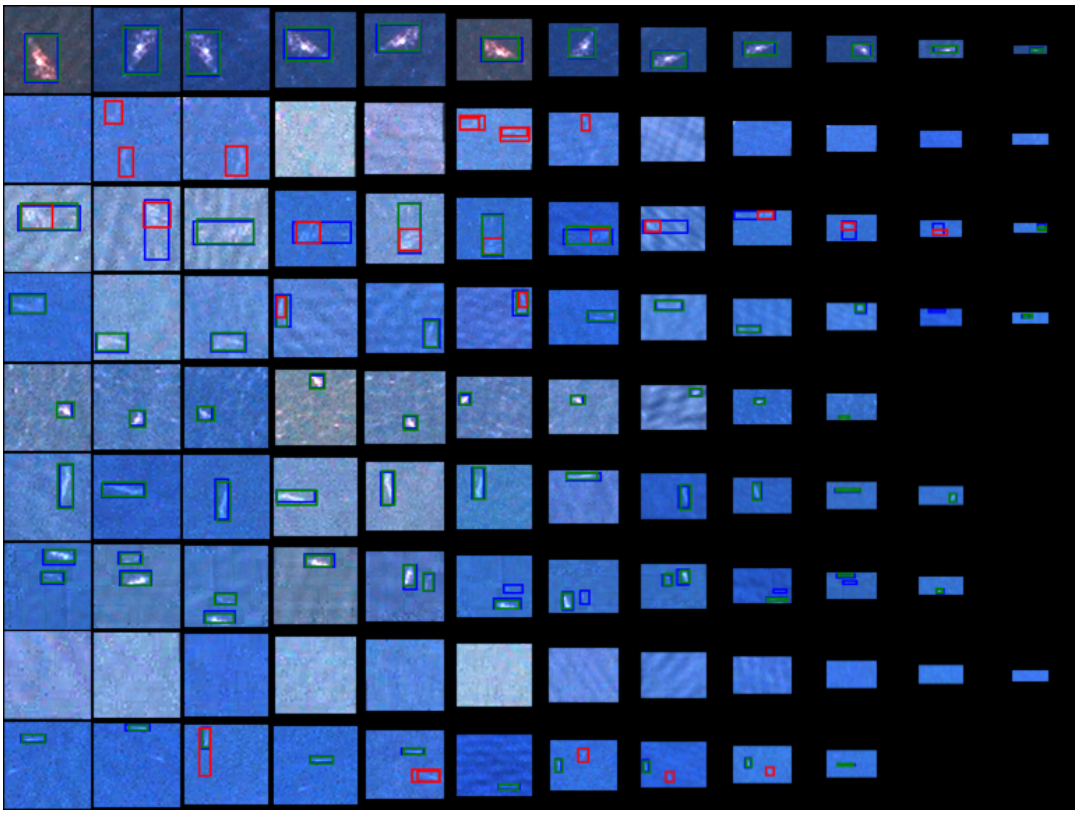


Figure 6.44: Detection examples on the Auckland2006 test set. Blue indicates ground truth, green indicates matched predictions (IoU> 0.5), and red indicates mismatched detections (IoU< 0.5). The model detects whales consistently across varying sizes and background conditions, with most errors related to bounding box extent rather than missed detections. *Satellite image © 2022 Maxar Technologies.*

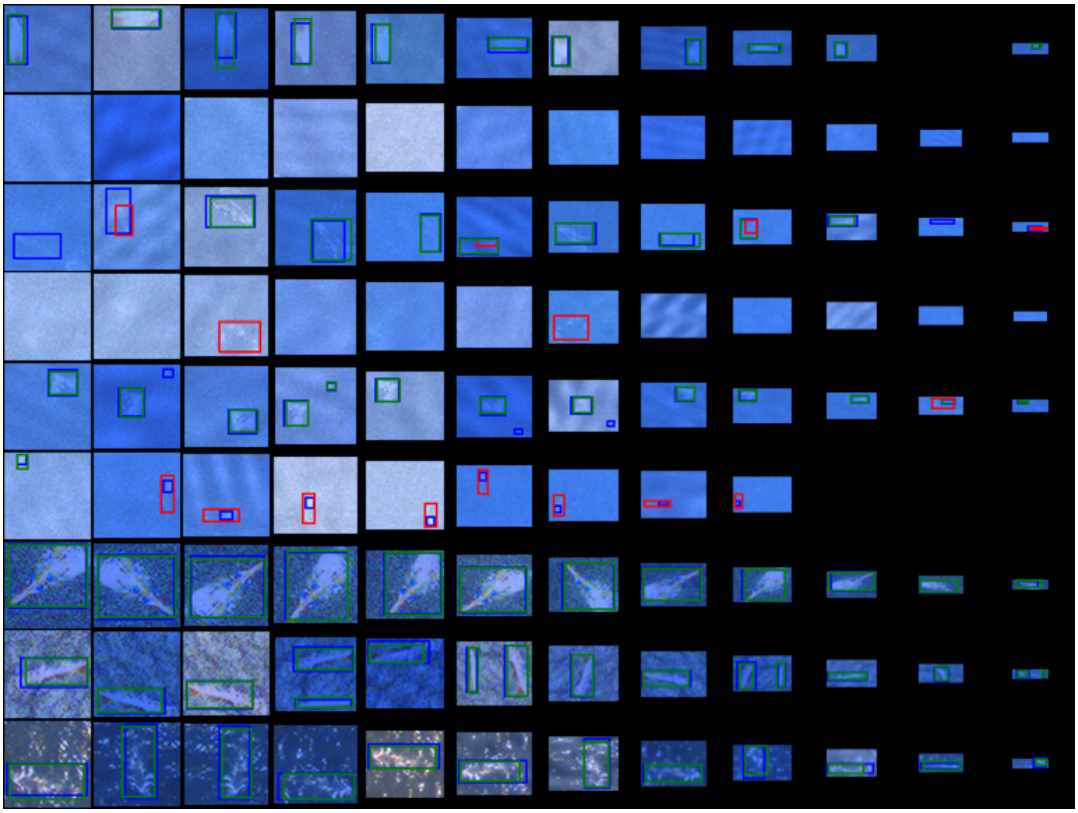


Figure 6.45: Detection examples on the Pelagos2016 test set. Blue indicates ground truth, green indicates matched predictions (IoU> 0.5), and red indicates mismatched detections (IoU< 0.5). The model adapts well to larger whale instances and maintains stable detection performance under varying radiometric conditions. *Satellite image © 2022 Maxar Technologies.*

Overall, it is observed that the model handles detections well across different whale shapes and sizes. In both Auckland2006 and Pelagos2016, whales are detected consistently until they become visually indistinguishable from the background, at which point the bounding boxes themselves become ambiguous. In the case of the large fin whale of approximately 22m in Pelagos2016, the model adapts well to the increased object size, despite this scale not being explicitly present in the training data. In addition, when multiple whales are present in the same patch, the model is generally able to distinguish and detect them as separate objects.

Radiometrically, the model is able to distinguish whales under varying illumination conditions. Even in the presence of sunglint, the model can still detect the whale object based on the remaining texture. Conversely, ocean-only regions with strong glint patterns are not consistently confused with whale objects.

Figure 6.46 shows a selection of demanding cases that are handled correctly. These include strong sunglint dominance, noisy backgrounds, small whales at high off-nadir angles, and whales that are only weakly visible in the ocean surface. The model is also able to distinguish whale objects from ocean surface structures, like fronts and elongated patterns.

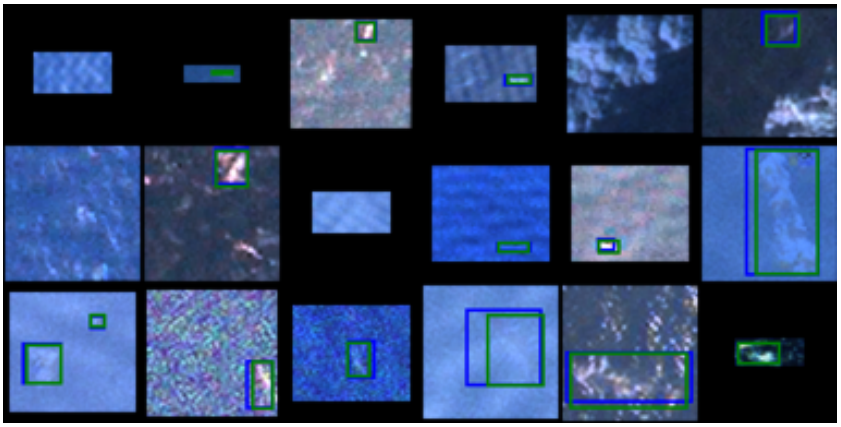


Figure 6.46: Examples of difficult cases that are correctly detected. These include strong sun glint, noisy ocean backgrounds, small whales at high off-nadir angles, and low-contrast targets. *Satellite image © 2022 Maxar Technologies.*

However, not all cases are handled perfectly. The main mismatch types observed are false positives, false negatives, and inaccuracies in bounding box size or placement. Among these, inaccuracies in bounding box extent are the most common.

Figure 6.47 shows examples of false positives, where ocean regions are incorrectly classified as whales. These errors typically occur in areas where the ocean surface exhibits patterns that resemble whale-like structures, but not in regions dominated by strong sunglint, which was initiall expected. They appear across both low and high off-nadir angles and often persist across multiple off-nadir angle samples of the same underlying patch.

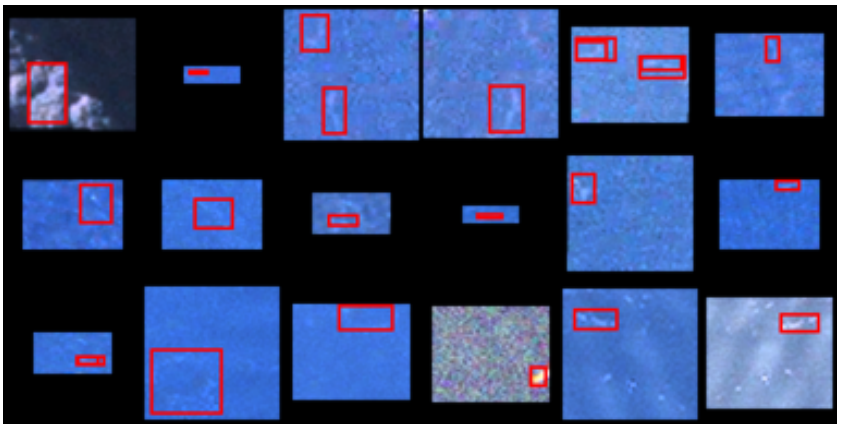


Figure 6.47: False positive detections. Red bounding boxes indicate predictions with IoU< 0.5. These errors typically occur in ocean regions with structures resembling whale-like shapes, rather than in areas dominated by strong sun glint. *Satellite image © 2022 Maxar Technologies.*

Figure 6.48 shows examples of false negatives. These typically occur when the whale becomes too small, too faint, or visually merges with surrounding ocean structures like wave fronts. These cases are more frequent at higher off-nadir angles, where the whales become smaller and contrast between the whale and the background decreases.

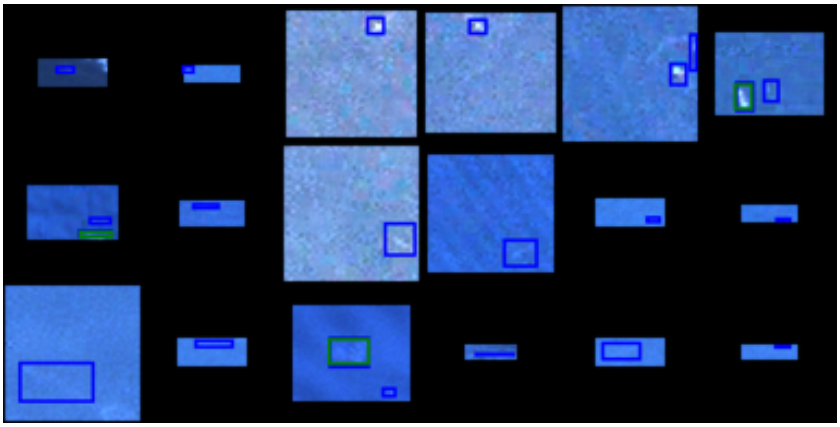


Figure 6.48: False negative examples. Blue indicates ground truth and green indicates matched predictions (IoU> 0.5). Missed detections occur primarily for small whales at high off-nadir angles or when whales visually merge with surrounding ocean structures. *Satellite image © 2022 Maxar Technologies.*

Lastly, inaccuracies in bounding box size and placement form the most common failure mode. These include cases where two nearby whales are merged into a single detection, or where the predicted bounding box does not correctly match the object extent. These mismatches occur more frequently for smaller whales and at higher off-nadir angles, where small localization errors have a larger impact on IoU given it is not scale-aware.

This confirms the use of AP_{50} as the primary evaluation metric. The initial bounding boxes are not fully precise, and the main objective is to detect the presence and approximate location of whales rather than their exact extent. A stricter metric, $AP_{50:95}$, would penalize small deviations in bounding box size more strongly, while these deviations are less relevant for the intended application.

In some cases, the correct bounding box is predicted, but a different, incorrect prediction with higher confidence is selected instead. To reduce those cases, future work could include additional samples containing fading whales and complex multi-object configurations. Furthermore, annotations with consistently inaccurate bounding boxes could be reverified before being used during training.

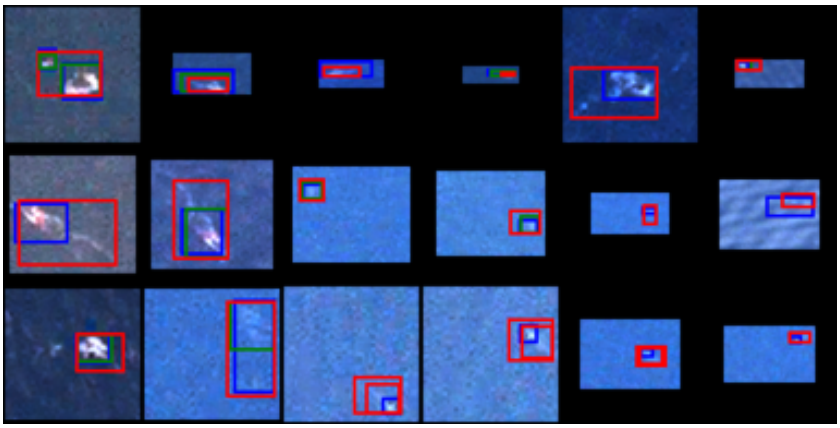


Figure 6.49: Examples of localization errors. Blue indicates ground truth, green indicates matched predictions (IoU> 0.5), and red indicates mismatched detections (IoU< 0.5). The dominant error mode is inaccurate bounding box extent, incorrect ordering of bounding box confidence scores, or merging of nearby whales into a single detection. *Satellite image © 2022 Maxar Technologies.*

6.3.4. Cross-Validation Split Variance

During the initial four-fold experiments, variation in validation performance across folds was observed. In particular, folds 3 and 4 consistently showed weaker performance than folds 1 and 2.

Because of this variance between folds during the hyperparameter optimization, it was desired to assess whether this results from data that is intrinsically more difficult, or from a generalization gap between

training and evaluation locations. This is also relevant for understanding whether the selection of the test locations affects the reported performance.

Therefore, to analyze the choice of the validation folds and test set, an individual experiment was set up in which the trained model was run on all locations separately, and AP_{50} metrics were extracted over the entire location run. It should be noted that the model used for this analysis was trained on 95% of the available non-test data and is therefore evaluated here on data that is no longer fully unseen. Therefore, this analysis is only focused on identifying differences in difficulty between the different input locations and folds.

Table 6.10 summarizes the final model performance on each of the nine locations.

Table 6.10: Performance per location, computed by running the optimized model across all locations. It should be noted that only Auckland2006 and Pelagos2016 belong to the test set. Therefore, the other locations are not fully unseen, and this table should only be used to examine variance between the folds, showing the relative difficulty with respect to each other.

	Dataset			Detection		Localization quality				F1-score	
Location	**Total**	**Whale**	**Non-whale**	**AP_{50}**	**$AP_{50:95}$**	**Avg. IoU**	**Std. IoU**	**Avg. conf.**	**Std. conf.**	**Best F1**	**Thr**
Auckland2006	1618	790	828	0.684	0.296	0.580	0.275	0.462	0.216	0.661	0.300
Auckland2011	769	361	408	0.877	0.440	0.702	0.197	0.504	0.155	0.778	0.300
Ignacio2017	1786	923	863	0.876	0.512	0.719	0.225	0.585	0.189	0.773	0.300
Maui2015	1162	646	516	0.808	0.463	0.705	0.236	0.541	0.185	0.679	0.300
Pelagos2016	674	374	300	0.887	0.495	0.738	0.183	0.599	0.163	0.778	0.300
Valdes2012	1895	911	984	0.849	0.435	0.700	0.211	0.545	0.177	0.743	0.300
Valdes2014	1308	685	623	0.964	0.619	0.804	0.139	0.674	0.144	0.854	0.300
Valdes2016	2910	1396	1514	0.928	0.513	0.768	0.125	0.604	0.112	0.806	0.300
Witsand2009	1803	975	828	0.888	0.430	0.718	0.156	0.579	0.128	0.803	0.300
Mean	1547	784	763	0.862	0.467	0.715	0.194	0.566	0.163	0.764	0.300
Std	697	351	347	0.082	0.090	0.061	0.050	0.061	0.031	0.067	0.000

Test Location Selection To evaluate the composition of the test set in Table 6.10, the test set consists of Auckland2006 and Pelagos2016, with AP_{50} values of 0.684 and 0.887, respectively. Scaling with the number of input images per location exactly gives the output test performance of 0.757 as evaluated from the training pipeline in subsection 6.3.2, which confirms that the evaluation pipeline has been set up correctly.

It shows that the low test score is mainly driven by Auckland2006, containing low-resolution WorldView-2 data, in contrast to Pelagos2016, which contains high-resolution WorldView-3 data and larger whales. Given that Pelagos reaches the same AP_{50} test scores despite being unseen, which has been reconfirmed by comparing the dumped COCO validation and test predictions where only the test set contained Pelagos2016, it is concluded that the model generalizes well to larger whales. It may even benefit from them, because they are more clearly visible at higher off-nadir angles. It is therefore expected that, if this location had been included in the training set, its AP_{50} scores would have been even higher.

To confirm this hypothesis, a training was performed on all locations, including Auckland2006 and Pelagos2016, after which the performance on the entire dataset was assessed to observe the relative difficulty on seen data. Table 6.11 shows the result.

Table 6.11: Performance per location obtained by running the optimized model separately on each of the nine locations after training on all available locations, including Auckland2006 and Pelagos2016. Since all locations are included in training, none of the reported results correspond to unseen-domain evaluation. This table is therefore only used to compare the relative difficulty of the locations under seen-data conditions.

	Dataset			Detection		Localization quality				F1-score	
Location	**Total**	**Whale**	**Non-whale**	**AP_{50}**	**$AP_{50:95}$**	**Avg. IoU**	**Std. IoU**	**Avg. conf.**	**Std. conf.**	**Best F1**	**Thr**
Auckland2006	1618	790	828	0.935	0.527	0.752	0.158	0.614	0.141	0.839	0.300
Auckland2011	769	361	408	0.941	0.502	0.738	0.170	0.572	0.156	0.842	0.300
Ignacio2017	1786	923	863	0.929	0.576	0.766	0.186	0.663	0.170	0.853	0.300
Maui2015	1162	646	516	0.904	0.557	0.763	0.188	0.639	0.174	0.785	0.300
Pelagos2016	674	374	300	0.959	0.653	0.808	0.152	0.724	0.124	0.891	0.300
Valdes2012	1895	911	984	0.906	0.492	0.741	0.177	0.614	0.164	0.831	0.300
Valdes2014	1308	685	623	0.984	0.678	0.839	0.100	0.738	0.119	0.909	0.300
Valdes2016	2910	1396	1514	0.969	0.594	0.798	0.113	0.688	0.106	0.905	0.300
Witsand2009	1803	975	828	0.928	0.481	0.748	0.132	0.634	0.113	0.872	0.300
Mean	1547	784	763	0.939	0.562	0.773	0.153	0.654	0.141	0.858	0.300
Std	697	351	347	0.025	0.066	0.034	0.028	0.053	0.023	0.037	0.000

It is indeed confirmed that Pelagos2016 is among the top three performers and its AP_{50} score rises after including its data in the training set. The same holds for Auckland2006, confirming that it is not necessarily the training data that is harder, but rather a generalization gap from the model to the Auckland2006 location. This has to be accounted for in the off-nadir evaluation.

Hyperparameter Optimization Folds Initially, during the hyperparameter optimization through cross-fold validation, there was a hypothesis that Valdes2016, which was the only location affected by the lack of preprocessing in the input data, exhibits different data characteristics that negatively impact performance and therefore lead to a generalization gap. However, Valdes2016 instead belongs to the top-performing locations when those samples are included in the training set, so this hypothesis had to be revised.

Therefore, it is more likely that the reduced fold performance is caused by the evaluation of this location in an unseen-domain setting, and that Valdes2016 data is too different to generalize to without having seen it.

To assess this, the final model configuration was retrained and evaluated over all $\binom{7}{2} = 21$ possible validation-set combinations of the seven non-test locations. The folds were then grouped according to whether Valdes2016 was included in the validation set, as shown in Table 6.12. Here, the folds containing Valdes2016 in validation correspond to an unseen-domain setting, where Valdes2016 is excluded from training in those folds. The full location splits are provided in section E.5.

Table 6.12: Grouping of folds based on the presence of Valdes2016 in the validation set. Folds where Valdes2016 is included correspond to an unseen-domain evaluation.

Group	**Folds**
Valdes2016 in validation (unseen)	3, 4, 9, 11, 17, 20
Valdes2016 not in validation	1, 2, 5, 6, 7, 8, 10, 12, 13, 14, 15, 16, 18, 19, 21

The resulting grouped performance is given in Table 6.13 and Table 6.14. A difference is observed on the validation set. When Valdes2016 is included in validation, the mean validation AP_{50} decreases from 0.771 to 0.682, corresponding to a drop of approximately 0.1. A similar reduction is observed for $AP_{50:95}$, which decreases from 0.361 to 0.284. This indicates that Valdes2016 forms a more challenging validation domain when it is not represented in the training data.

Table 6.13: Performance for folds where Valdes2016 is included in the validation set (unseen domain setting). Metrics are reported separately for the validation and test sets per fold.

		Validation				Test			
Fold	**Validation locations**	**AP_{50}**	**$AP_{50:95}$**	**Best F1**	**Thr**	**AP_{50}**	**$AP_{50:95}$**	**Best F1**	**Thr**
fold3	Valdes2014, Valdes2016	0.711	0.316	0.928	0.422	0.738	0.347	0.928	0.328
fold4	Valdes2012, Valdes2016	0.665	0.275	0.922	0.392	0.720	0.329	0.928	0.351
fold9	Auckland2011, Valdes2016	0.683	0.276	0.920	0.384	0.727	0.342	0.927	0.318
fold11	Ignacio2017, Valdes2016	0.694	0.303	0.928	0.382	0.717	0.345	0.929	0.328
fold17	Valdes2016, Witsand2009	0.689	0.268	0.938	0.394	0.736	0.349	0.928	0.343
fold20	Maui2015, Valdes2016	0.649	0.268	0.903	0.374	0.723	0.344	0.927	0.321
Mean		**0.682**	**0.284**	**0.923**	**0.391**	**0.727**	**0.343**	**0.928**	**0.331**

Table 6.14: Performance for folds where Valdes2016 is not included in the validation set. Metrics are reported separately for the validation and test sets per fold.

		Validation				Test			
Fold	**Validation locations**	**AP_{50}**	**$AP_{50:95}$**	**Best F1**	**Thr**	**AP_{50}**	**$AP_{50:95}$**	**Best F1**	**Thr**
fold1	Auckland2011, Witsand2009	0.799	0.335	0.966	0.233	0.756	0.360	0.929	0.210
fold2	Ignacio2017, Maui2015	0.703	0.331	0.921	0.331	0.721	0.348	0.928	0.340
fold5	Auckland2011, Valdes2014	0.850	0.435	0.945	0.343	0.748	0.356	0.932	0.197
fold6	Ignacio2017, Witsand2009	0.764	0.347	0.958	0.345	0.739	0.363	0.930	0.322
fold7	Maui2015, Valdes2012	0.722	0.325	0.924	0.269	0.732	0.351	0.929	0.189
fold8	Valdes2012, Valdes2014	0.802	0.401	0.943	0.337	0.743	0.356	0.932	0.273
fold10	Maui2015, Witsand2009	0.740	0.318	0.940	0.301	0.744	0.366	0.931	0.285
fold12	Auckland2011, Ignacio2017	0.754	0.352	0.937	0.333	0.752	0.359	0.936	0.240
fold13	Maui2015, Valdes2014	0.772	0.391	0.907	0.358	0.745	0.354	0.928	0.280
fold14	Valdes2012, Witsand2009	0.786	0.346	0.960	0.293	0.759	0.367	0.931	0.322
fold15	Auckland2011, Maui2015	0.699	0.306	0.912	0.325	0.740	0.350	0.931	0.209
fold16	Ignacio2017, Valdes2014	0.822	0.421	0.937	0.413	0.734	0.350	0.925	0.284
fold18	Auckland2011, Valdes2012	0.761	0.355	0.950	0.205	0.739	0.353	0.927	0.189
fold19	Valdes2014, Witsand2009	0.843	0.395	0.958	0.363	0.755	0.363	0.933	0.294
fold21	Ignacio2017, Valdes2012	0.746	0.351	0.943	0.316	0.733	0.354	0.934	0.305
Mean		**0.771**	**0.361**	**0.940**	**0.318**	**0.743**	**0.357**	**0.931**	**0.263**

At the same time, the average test performance remains nearly unchanged. The mean test AP_{50} is 0.727 for folds with Valdes2016 in validation and 0.743 for folds without it, while $AP_{50:95}$ differs only marginally, from 0.343 to 0.357. Therefore, the Valdes2016 effect appears localized to the validation domain.

A secondary effect is visible in the best F1 threshold, which is higher for the folds containing Valdes2016 in validation, equal to 0.391 vs 0.318. This is consistent with the lower original intensity distribution of Valdes2016 discussed earlier, which may shift the confidence calibration required for optimal classification.

Finally, when all available non-test data is pooled and 95% is used for training, the final model reaches an AP_{50} of 0.894 on validation and 0.757 on test, as shown in Table 6.9. Together with the strong per-location performance on Valdes2016 in Table 6.10, this indicates that Valdes2016 is a learnable domain and that including it in the final training set is beneficial for generalization. The earlier weak performance of folds 3 and 4 should therefore be interpreted primarily as an unseen-domain validation effect, while Valdes2016 still remains suitable for the model.

Overall, the results indicate that the observed variation between folds is largely driven by the specific choice of validation and test locations, in particular whether Valdes2016 is included or not. While the mean test score of 0.757 and the mean cross-fold test scores of 0.727-0.743 are closely aligned, this suggests that the selected test set is representative of the overall performance across folds.

However, a strong difference is observed between the individual test locations, with Auckland2006 achieving an AP_{50} of 0.684 compared to 0.887 for Pelagos2016. This indicates a generalization gap for Auckland2006, while Pelagos2016 may even be comparatively easier due to the larger whales. This generalization gap should be taken into account in further analysis, where lower performance on Auckland2006

should not be attributed solely to its larger ground sampling distance, but also to the limited generalization of the trained model to this location.

6.4. Integrated Pipeline Demonstration

This section demonstrates the integrated end-to-end pipeline. Since no actual AI-Based Tip and Cue mission currently exists for direct mission-level validation, the objective here is to show that the verified components operate consistently once coupled in one simulation chain, from Tip detection and Cue tasking to off-nadir image generation and final onboard confirmation.

6.4.1. Baseline Demonstration

In the baseline scenario, a maximum off-nadir angle of 40 degrees and a time delay of 5 minutes between Tip and Cue are imposed. A full simulation was executed, resulting in the orbital evolution shown in Figure 6.50, also available at: `https://youtu.be/RVM3C2Z773Q`.

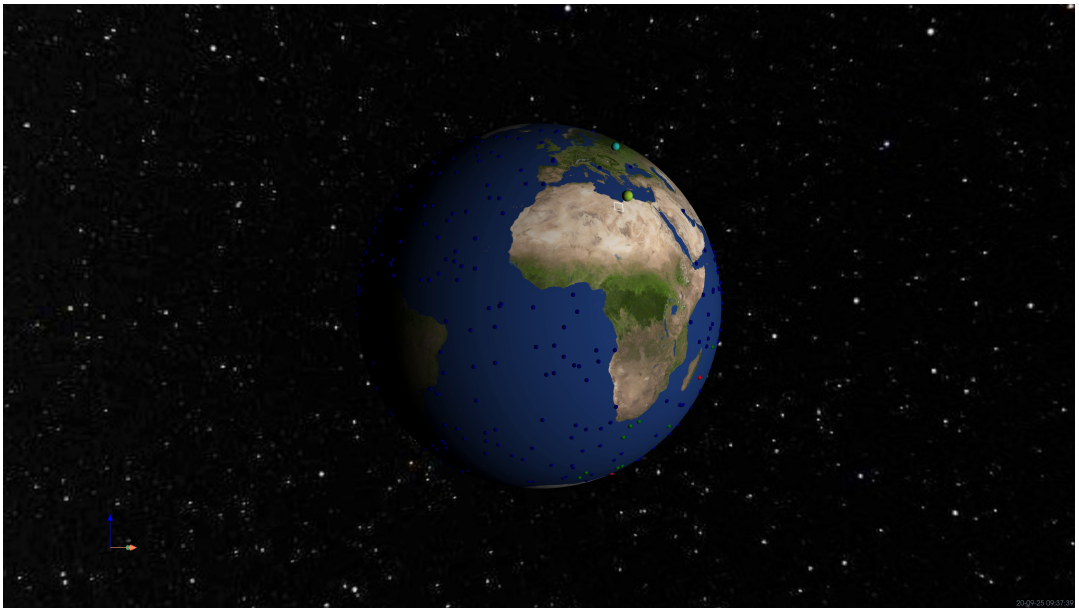


Figure 6.50: Orbital simulation snapshot showing the global distribution of targets (blue), detection outcomes (true positives in green and false negatives in red), and satellite positions during the simulation. The interactive PyVista visualization is updated in real time, while in automated workflows it is hidden and outputs are recorded.

At 20 September 2025 at 12:51:59, a target is detected by the Tip satellite in the North Atlantic Ocean, approximately at the latitude of Morocco. The Cue satellite observes the same target at 12:55:19, matching the imposed 5-minute delay while accounting for forward-pointing capability, as illustrated in Figure 6.51.

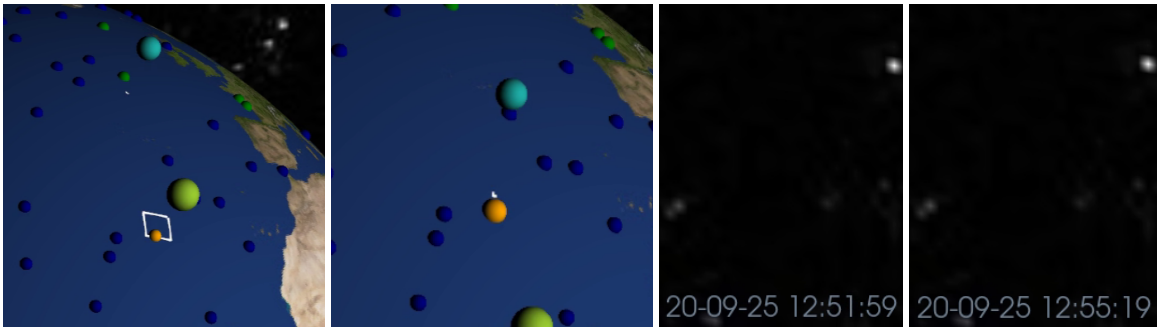


Figure 6.51: Detection sequence showing Tip detection, Cue observation, and corresponding timestamps validating the imposed time delay.

The observation is recorded at an off-nadir angle of 39.88 degrees, satisfying the imposed constraint of 40 degrees while minimizing latency.

After the Cue satellite observes target 261, a universally unique identifier (UUID): e20f7d87-67b7-4641-b5f9-02969241d82f is assigned. A radiometrically consistent image is then generated, as shown in Figure 6.52.

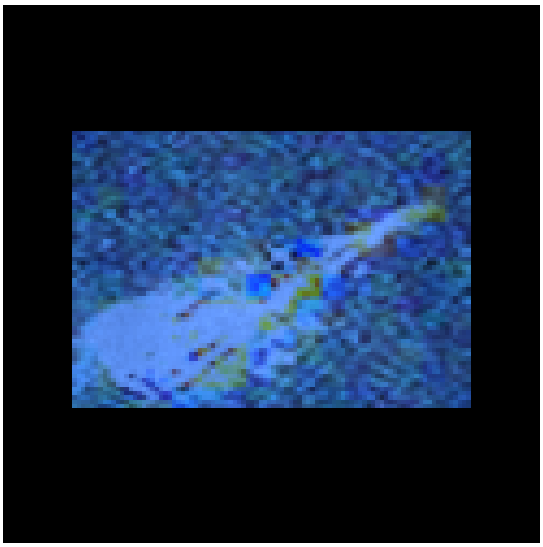


Figure 6.52: Generated radiometrically consistent satellite image corresponding to the Cue observation. *Satellite image © 2022 Maxar Technologies.*

Subsequently, the optimized DEIMv2 onboard detection model is deployed, successfully identifying the target, as shown in Figure 6.53.

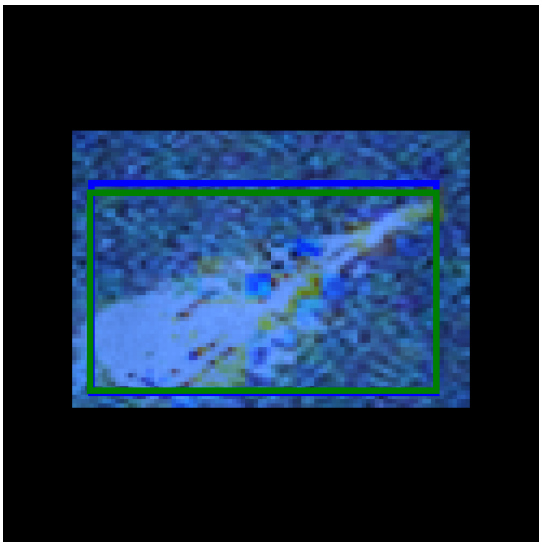


Figure 6.53: DEIMv2 onboard detection result with predicted bounding box overlaid on the generated image. *Satellite image © 2022 Maxar Technologies.*

The generated satellite image and its annotation are stored in `satellite_images`. Corresponding radiance and reflectance data are stored in `supporting_dataset`. Prediction outputs are saved in `prediction_images`, and detection statistics are recorded in `onboard_detection_results.xlsx`. The images are shown in Figure 6.54.

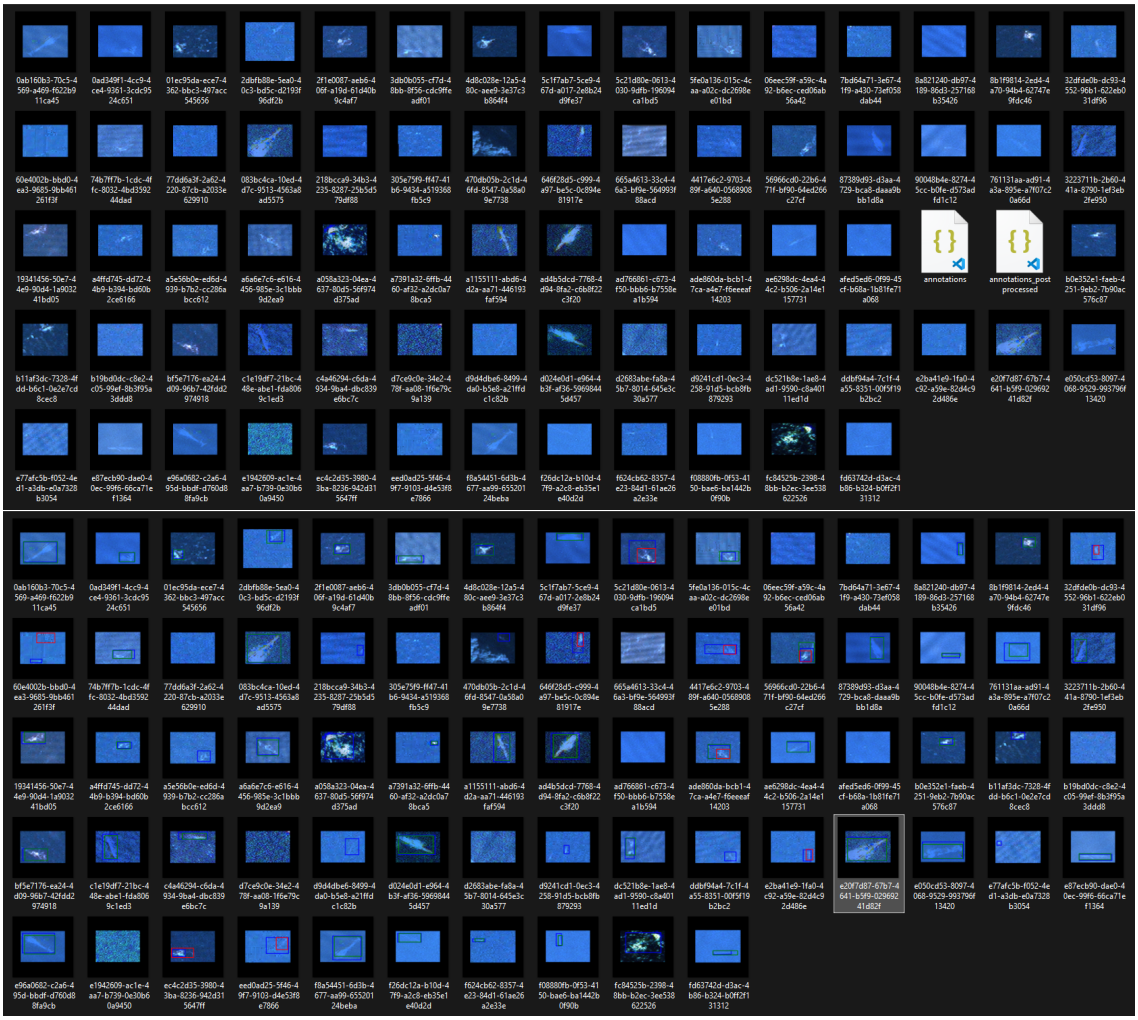


Figure 6.54: Generated Cue satellite image dataset (top) and corresponding onboard detection outputs with predicted bounding boxes (bottom). Each sample has a unique image identifier which can be traced back in the annotations, radiance/reflectance data, and detection statistics, which demonstrates the complete end-to-end processing pipeline. *Satellite image © 2022 Maxar Technologies.*

At the end of the simulation, the image folders contain forty images, corresponding to the total number of Cue observations of this particular run.

Global coverage and access patterns are provided through footprint maps (`footprints_tip.html` and `footprints_cue.html`), shown in Figure 6.55. Additionally, a simulation video (`mov_<run_name>.mov`) is generated in 4K UHD resolution.

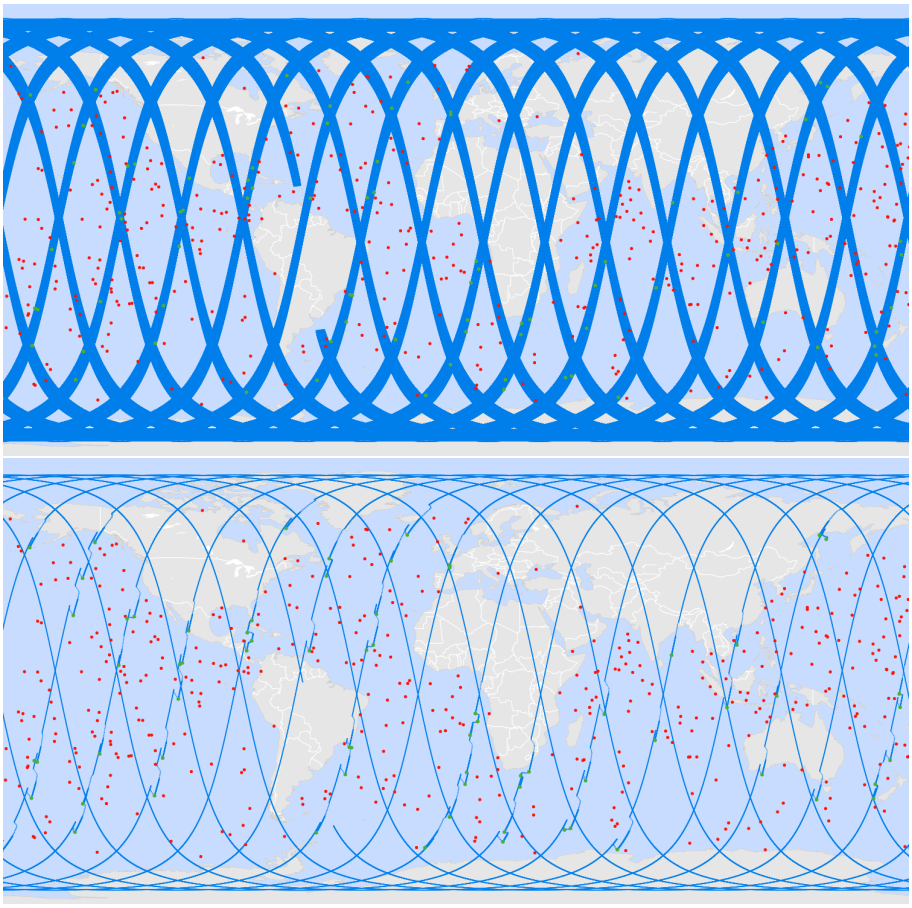


Figure 6.55: Global ground tracks and observation footprints for Tip (top) and Cue (bottom) satellites. The outputs are provided as interactive `.html` files, enabling panning and zooming.

Finally, a `benchmark.xlsx` file is generated summarizing the overall performance metrics of the simulation, as described in subsection 5.1.1.

For reproducibility and debugging, terminal output logs, observation events including time and location, configuration settings, and satellite parameters are stored.

6.4.2. Customization Options

The simulation framework supports flexible configuration. Both constellation-based setups and individually defined satellites can be specified.

Configurable parameters include orbital elements, number of satellites, number of orbital planes, inter-satellite spacing, Tip-Cue delay, maximum off-nadir angle, and target dynamics.

Additional options include swath width, satellite image rendering resolution, attitude control dynamics, and configurable detection performance, specifying probabilistic detection rates when no model is used.

Example configurations are shown in Figure 6.56, demonstrating different constellation layouts and operational constraints.

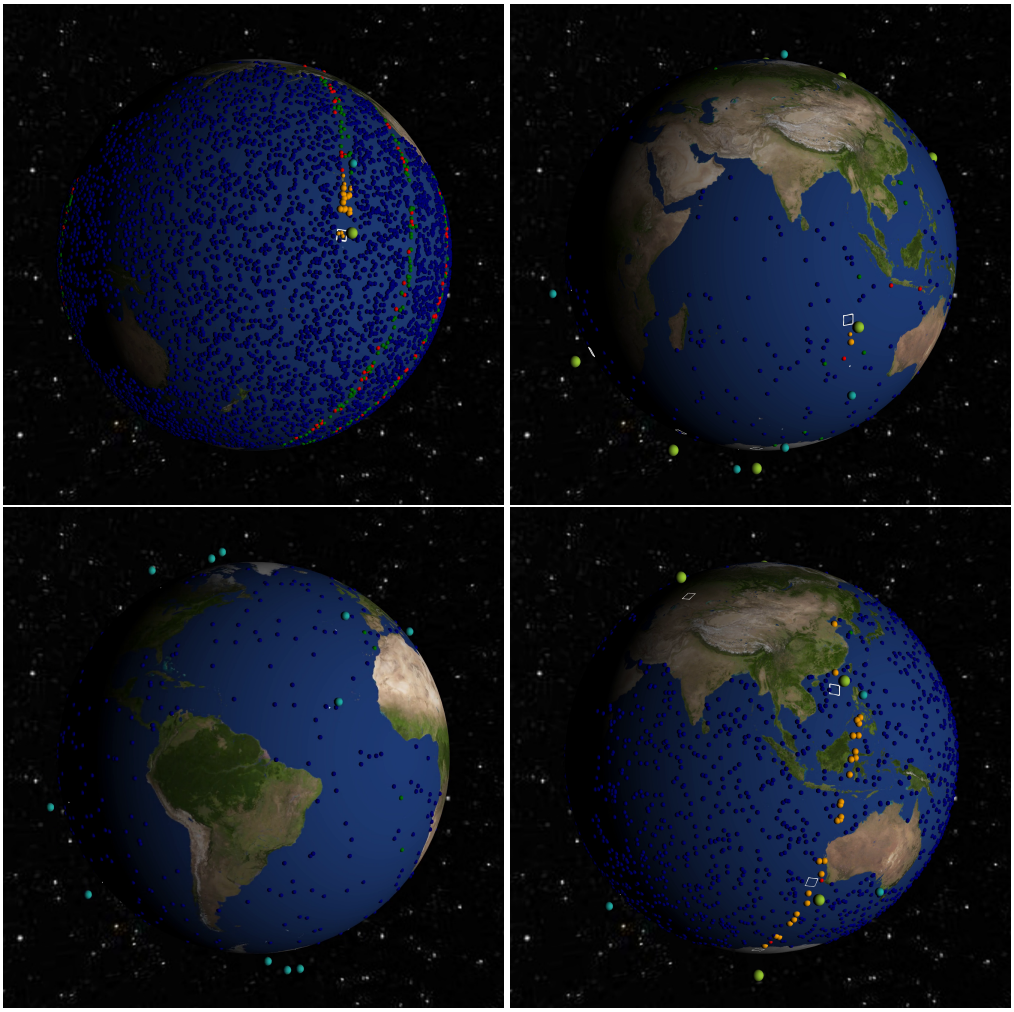


Figure 6.56: Example simulation configurations demonstrating different system architectures and operational modes. From top left to bottom right: (1) single Tip-Cue system with 10,000 targets, (2) 4x2 Tip-Cue constellation, (3) 16-satellite VHR constellation, and (4) 12 independently operating satellites. In the independent case, targets are tasked but no coordinated Cue follow-up is performed.

The AI-Based Tip and Cue simulation framework is available as open-source software at `https://github.com/ESA-PhiLab/AI-Based-Tip-and-Cue`.

7

Results

This chapter presents the results of the integrated AI-Based Tip and Cue evaluation. First, subsection 7.1.1 and subsection 7.1.2 assess how mission geometry affects accessibility, latency, viewing time, and successful confirmation performance. Next, section 7.2 evaluates how representative geometric and radiometric training data affect Cue-side detection performance. Finally, section 7.3 compares AI-Based Tip and Cue against alternative Earth Observation architectures under the defined benchmark.

7.1. Mission Geometry

This section evaluates how mission geometry affects end-to-end AI-Based Tip and Cue performance. Following the benchmark, it quantifies the effect of the maximum allowable Cue off-nadir angle and the time delay between Tip and Cue on accessibility, latency, viewing time, and successful confirmation performance. First, subsection 7.1.1 evaluates the off-nadir angle sweep, after which subsection 7.1.2 assesses the time-delay sweep.

7.1.1. Off Nadir Limit

To evaluate the off-nadir limit of the detection model was evaluated, the dataset was first visually explored to observe when whales vanish in their oceanic surroundings, after which the operational off-nadir limit of the model was assessed through the mission simulations.

Visual Evaluation

Examples of generated images over the off-nadir range from $10°$ to $60°$ are shown for the low and high ground sampling distance groups in Figure 7.1 and Figure 7.2, respectively.

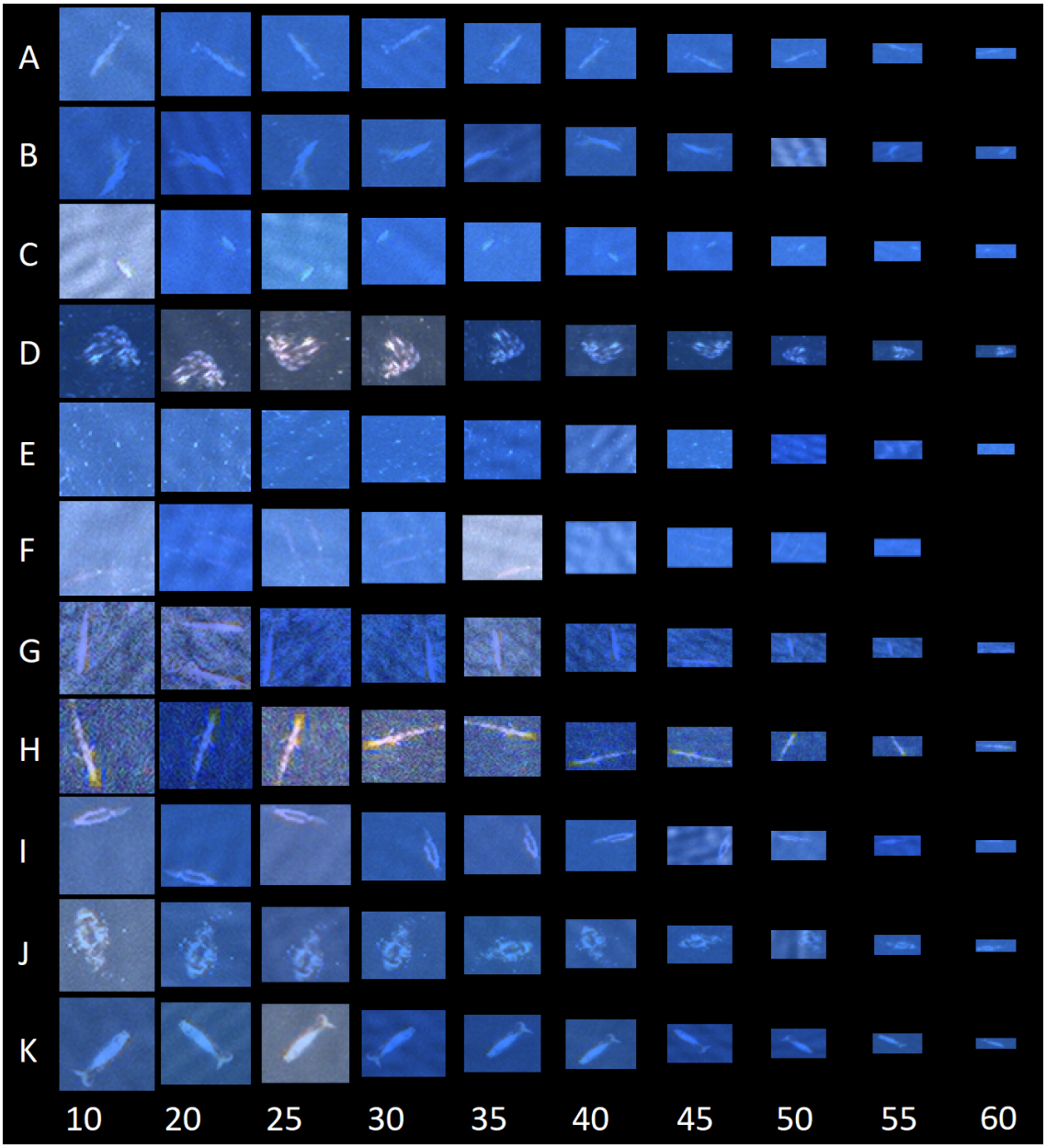


Figure 7.1: Examples of generated images at 10°, 20°, 25°, 30°, 35°, 40°, 45°, 50°, 55°, and 60° off-nadir for the low ground sampling distance group. From top to bottom: A-C: Ignacio2017, D-F: Maui2015, G-I: Pelagos2016, J-K: Valdes2014. Row E represents ocean only, while the remaining rows contain whales. *Satellite image © 2022 Maxar Technologies.*

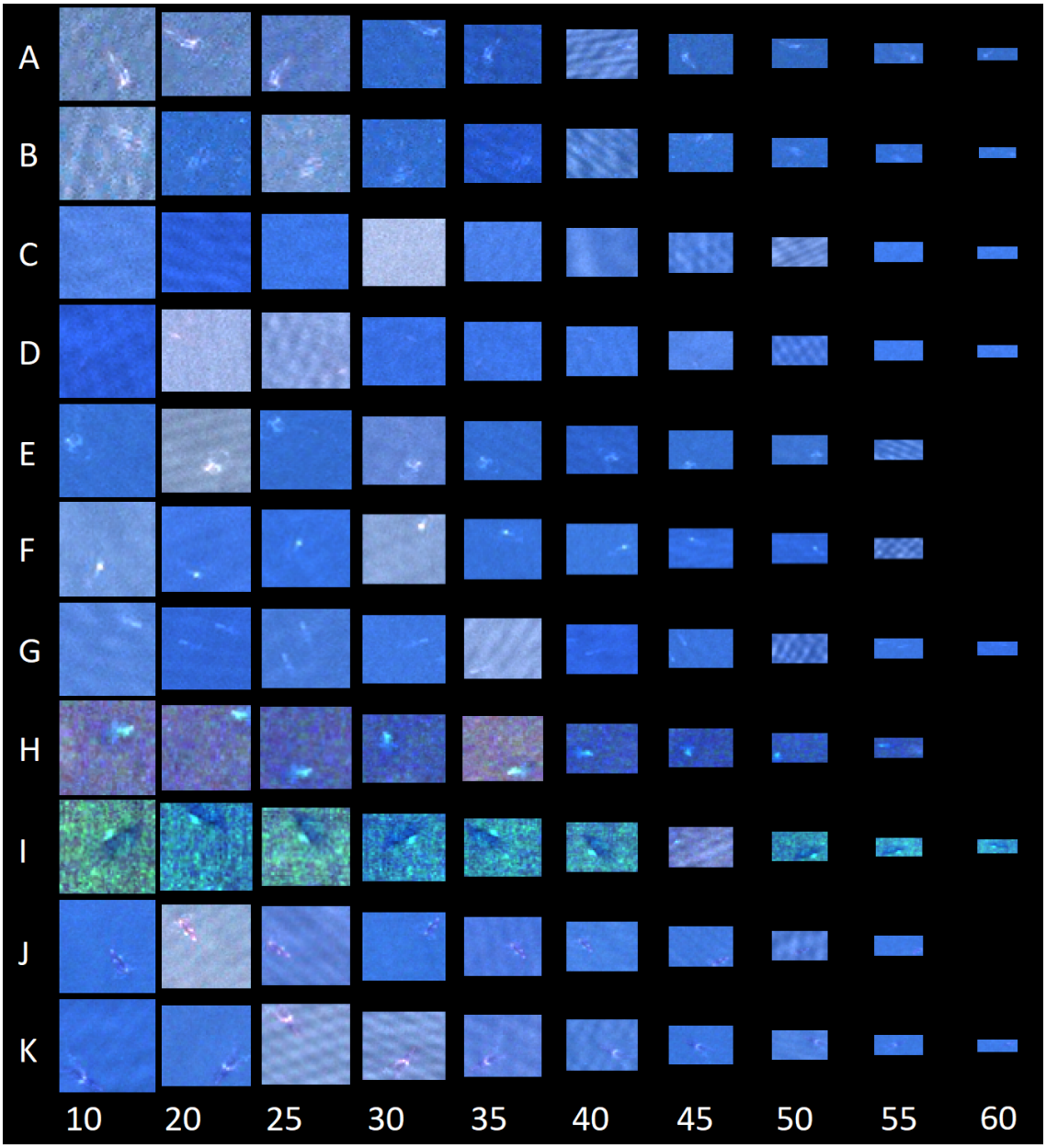


Figure 7.2: Examples of generated images at 10°, 20°, 25°, 30°, 35°, 40°, 45°, 50°, 55°, and 60° off-nadir for the high ground sampling distance group. A-B: Auckland2006, C-D: Auckland2011, E-G: Valdes2012, H-I: Valdes2016, J-K: Witsand2009. Row C represents ocean only, while the remaining rows contain whales. Rows H and I are from Pelagos2016 and are the only ones affected by missing normalization, as discussed in subsection 6.2.7. *Satellite image © 2022 Maxar Technologies.*

First, some images at 60° are missing. This occurs when the whale becomes so small that the bounding-box corner points overlap and no annotation can be generated, as described in subsection 5.4.1. In that case, the detector would not be able to observe the whale either, and the sample is therefore counted as missed. As expected, this occurs more often for the high ground sampling distance group, because the whales are smaller and thus vanish sooner.

Comparing both image boards, the whales in the low ground sampling distance group generally remain visible over a larger off-nadir range. An exception is row F in Figure 7.1, where the whales are more submerged and vanish faster into the background at larger off-nadir angles.

In Figure 7.2, the whales typically vanish faster in their oceanic surroundings. Up to approximately 40° to 45°, they remain clearly visible, after which their intensity reduces and they become harder to distinguish from the background.

From approximately 40° onwards, sun glint effects, where present, become more dominant. Ocean glint may be misclassified as whale, for example at C45, C50, E55, and F55 in Figure 7.2. Likewise, whales may be missed while disappearing into sun glint, as can be observed at B50 and F40 in Figure 7.1, and at A40, B40, D50, G50, and J50 in Figure 7.2.

Based on these observations, the model is expected to operate reliably up to around 55° for the low ground sampling distance patches, and up to around 45° for the high ground sampling distance patches. Beyond that, performance is expected to gradually degrade. Still, this does not depend on merely ground sampling distance, while also the geometry and lighting conditions of the original texture image should be considered, which vary across the nine location datasets.

Operational Limit

Next, it was quantified up to which off-nadir angles the model remains robust in the mission simulations. To do so, the results were evaluated separately for the test locations Pelagos2016 and Auckland2006, thus for both the low and high ground sampling distance groups.

To evaluate the effect of the maximum off-nadir requirement, the outputs of the three simulation runs were combined for each configuration covering maximum off-nadir angles from 5° to 60°, with a step size of 5°. The detector was then deployed on the generated Pelagos2016 and Auckland2006 data of each mission.

After that, each configuration was evaluated following the benchmark in terms of successful detection capacity, handled tasks, latency, viewing time, and detector performance over the off-nadir range.

Mission Metrics The successful detection capacity per satellite, both mean and across runs, is shown in Figure 7.3. Note that the benchmark reports detections per satellite, so the total number of successful detections for the full mission is twice that amount.

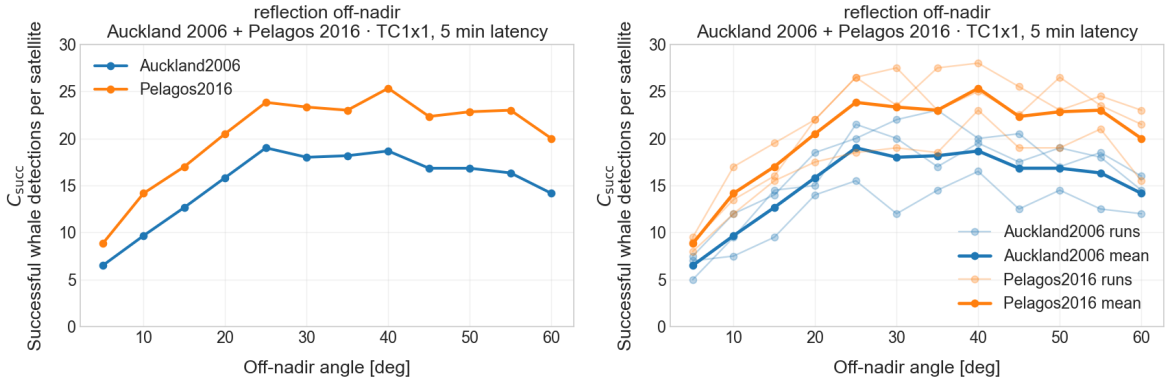


Figure 7.3: Successful detection capacity per satellite, $C_{\text{succ,satellite}}$, as a function of maximum allowed off-nadir angle for the TC1x1 configuration with a 5 minute Tip-Cue delay, evaluated on Auckland2006 and Pelagos2016. Left: mean across runs. Right: individual runs and mean.

First, the successful detection capacity varies across runs. This indicates that the result depends on whale seed, which determines both the number of observation opportunities and which whale images are propagated to the final dataset. This becomes visible at 40°, where the Pelagos peak is caused by one favorable run. Given this variance, either the number of targets should be increased beyond 500, or more runs across whale seeds should be added.

To separate whale-seed variation from detector behavior, Figure 7.4 shows the number of Cue tasks handled across the same off-nadir sweep, independent of whether the detector succeeded.

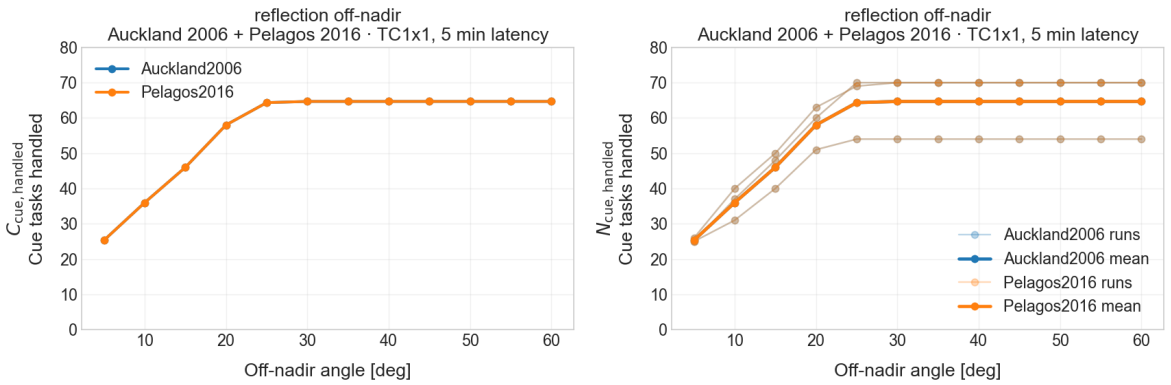


Figure 7.4: Cue-task handling as a function of maximum allowed off-nadir angle for the TC1x1 configuration with a 5 minute Tip-Cue delay, evaluated on Auckland2006 and Pelagos2016. Left: mean handled task count in benchmark form. Right: individual runs and mean.

Figure 7.4 confirms that different whale seeds can change the number of handled tasks by approximately ± 10 across runs, but do not explain the local fluctuations within one run. Those fluctuations therefore have to be attributed to detector performance.

The increase in successful detection capacity up to approximately $25°$ is mainly caused by the fact that more tasks can be executed up to that point. At small maximum off-nadir angles, not every lateral position inside the Tip window can be reached by the Cue satellite, which leads to fewer detections overall. In the case of Pelagos2016, $C_{\text{succ,satellite}}$ increases from roughly 9 to roughly 24, thus by about 15 detections per satellite, while the total number of handled Cue tasks increases from roughly 32 to roughly 68. Therefore, the main increase up to this point is caused by improved task reachability, while detector performance remains relatively stable.

From $25°$ onwards, the two locations behave differently. For Auckland2006, performance already starts to decline with increasing off-nadir angle. Pelagos2016 continues to increase up to around $40°$, after which the drop is observed again from $55°$ onwards. This is consistent with the visual observations in subsection 7.1.1: Auckland2006 contains smaller whales that vanish faster with increasing off-nadir angle, whereas Pelagos2016 contains larger whales that remain visible for longer and continue to benefit from the larger sideways, forward, and backward pointing range.

It is also observed that between $25°$ and $40°$, both curves initially start to drop, after which a local peak appears around $40°$. Since the number of handled tasks has already saturated by approximately $25°$, this increase is not caused by additional reachability. A likely explanation is the black border that appears around the image patches at higher off-nadir angles. As long as the whale remains visible, the effective search area becomes smaller, which can make the detection task slightly easier. This is also reflected by the small increase in AP_{50} and recall between approximately $35°$ and $45°$.

Response Metrics The response metrics latency and viewing time are shown in Figure 7.5. The variance across runs is small, because these are geometric effects and the mission geometry is fixed across whale seeds.

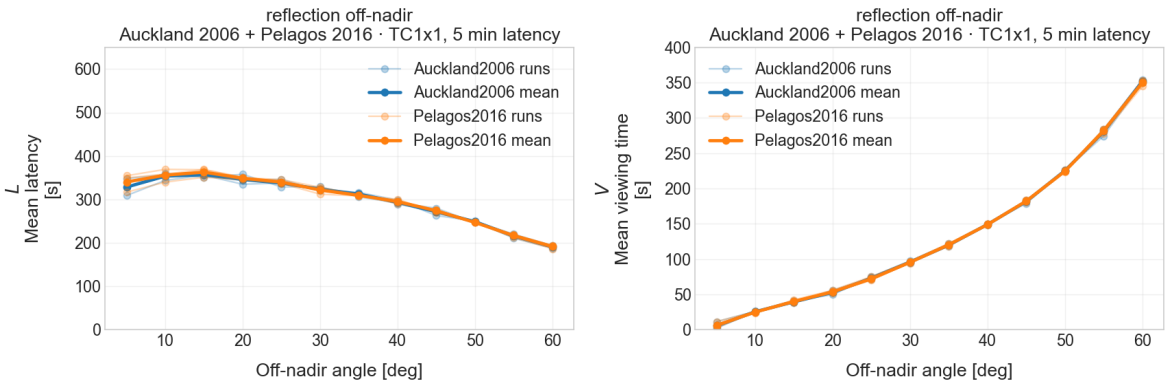


Figure 7.5: Response metrics as a function of maximum allowed off-nadir angle for the TC1x1 configuration with a 5 minute Tip-Cue delay, evaluated on Auckland2006 and Pelagos2016. Left: mean latency of successful observations, L. Right: mean viewing time of successful observations, V.

With increasing off-nadir angle beyond approximately $15°$, the latency decreases. This is expected, because the Cue satellite can point further ahead. Before $15°$, the latency increases. At very small off-nadir angles, not all targets can be reached, and latency is only computed for the successful cases. Those successful cases are biased toward geometrically favorable targets that can be reached with relatively little slewing. Balancing forward pointing capability, reach, and pointing time, the latency peak occurs around $15°$, after which lower latencies become possible. At $60°$, the mean latency decreases to approximately 190 seconds, for a Tip-Cue delay of 300 seconds and a Tip processing time of 90 seconds.

Figure 7.5 also shows that the mean viewing time increases strongly with off-nadir angle. This is expected, because with a larger off-nadir angle, the Cue satellite can point further ahead and observe the target earlier, and can also point backward for longer after passing nadir. The increase is steeper than linear, because the accessible ground range grows nonlinearly with viewing geometry.

Detection Metrics The detector metrics are shown in Figure 7.6. Pelagos2016 achieves higher AP_{50} scores than Auckland2006, as also discussed earlier in subsection 6.3.4. Up to approximately $45°$, Pelagos2016 shows a gradual decrease with a local kink around $35°$, while Auckland2006 shows local kinks around $20°$ and $35°$. As discussed above, these are likely caused by the black border around the high off-nadir patches, which reduces the area where the detector has to search and can therefore slightly increase performance artificially as long as the whale remains visible.

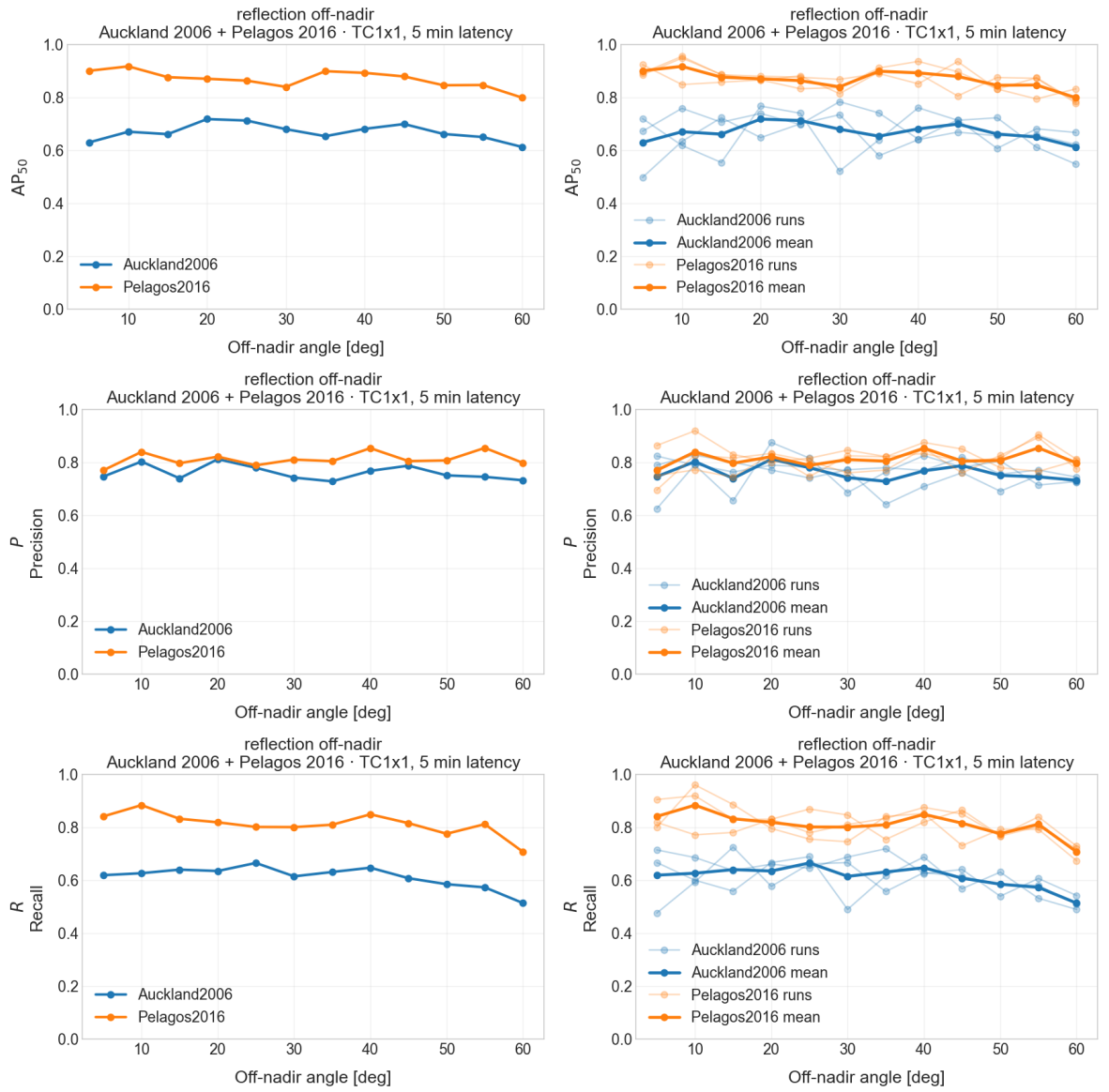


Figure 7.6: Detection metrics as a function of maximum allowed off-nadir angle for the TC1x1 configuration with a 5 minute Tip-Cue delay, evaluated on Auckland2006 and Pelagos2016. From top to bottom: AP_{50}, precision, and recall. Left: mean across runs. Right: individual runs and mean.

Overall, precision remains approximately constant, even between Auckland2006 and Pelagos2016. This means that, for the chosen detection threshold, the false-positive rate remains relatively stable across off-nadir angle and ground sampling distance. Recall, in contrast, shows a clearer drop beyond approximately $40°$. At higher off-nadir angles, more whales are missed, which is consistent with the visual observations in subsection 7.1.1, where increased effective ground sampling distance and stronger sun glint make whales harder to detect.

Overall, the results indicate that increasing Cue pointing capability is beneficial up to approximately $40°$, balancing viewing time, latency, and detection performance. However, part of the increase up to $40°$ is affected by the black-border artifact, which reduces the effective detection area and can artificially improve detector performance. In practice, the most robust detection regime is therefore expected closer to approximately $25°$, while the larger off-nadir angles mainly benefit latency and viewing time.

Depending on mission requirements, the maximum allowable latency and the maximum allowable off-nadir angle for detector performance therefore have to be balanced jointly.

7.1.2. Time Delay

Next, the effect of varying the time delay between the Tip and Cue satellite was evaluated.

Mission Metrics The successful detection capacity, both mean and across runs, is shown in Figure 7.7. The successful detection capacity is largely independent of the time delay between the Tip and Cue satellite. The scores vary by approximately ± 10 per location across the delay range, which matches the variance across different whale seeds.

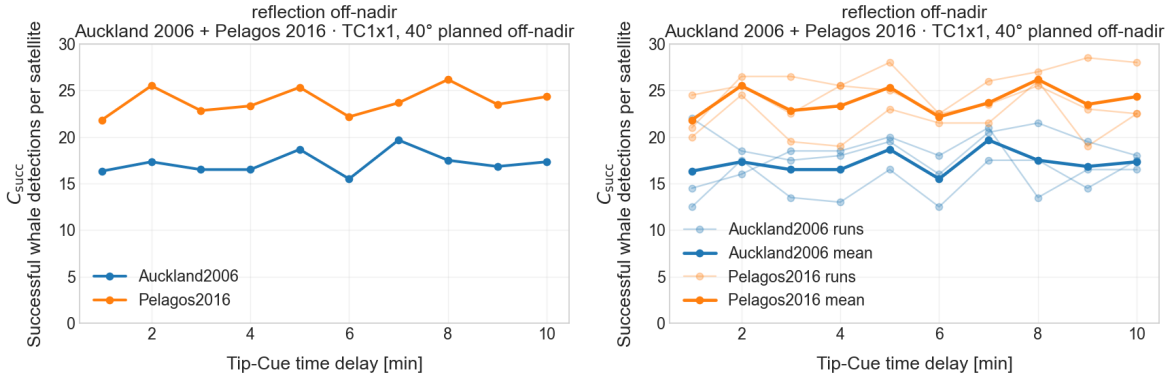


Figure 7.7: Successful detection capacity per satellite, $C_{\text{succ,satellite}}$, as a function of Tip-Cue time delay for the TC1x1 configuration with a maximum allowed off-nadir angle of $40°$, evaluated on Auckland2006 and Pelagos2016. Left: mean across runs. Right: individual runs and mean.

Part of this variation is caused by the initial whale seed, again indicating that either the number of targets or the number of runs should be increased for a less variant result. The remaining difference between runs is related to the number of tasks handled, as shown in Figure 7.8. Because the Tip satellite is shifted with respect to the fixed Cue position, some delay settings lead to a more favorable geometry in a particular run.

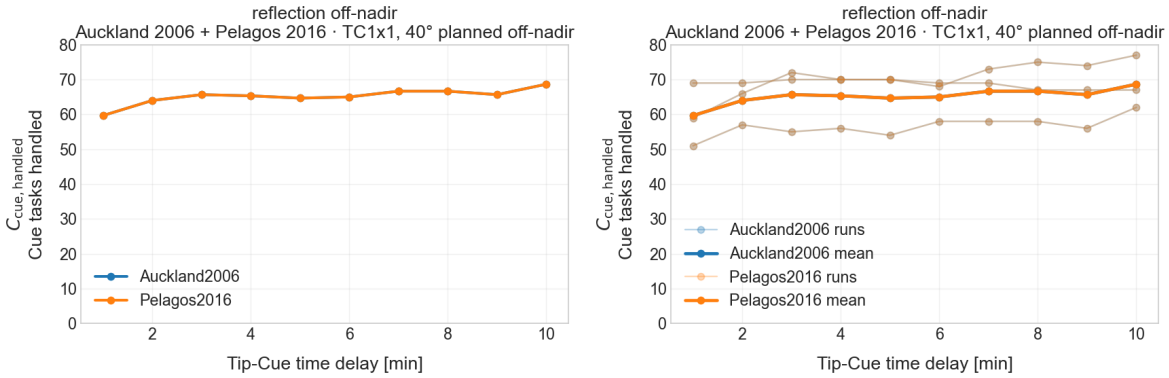


Figure 7.8: Cue-task handling as a function of Tip-Cue time delay for the TC1x1 configuration with a maximum allowed off-nadir angle of $40°$, evaluated on Auckland2006 and Pelagos2016. Left: mean handled task count in benchmark form. Right: individual runs and mean.

While the number of handled tasks increases slightly with time delay, the number of successful detections remains approximately stable. To explain this, the mean observed off-nadir angle is shown in Figure 7.9.

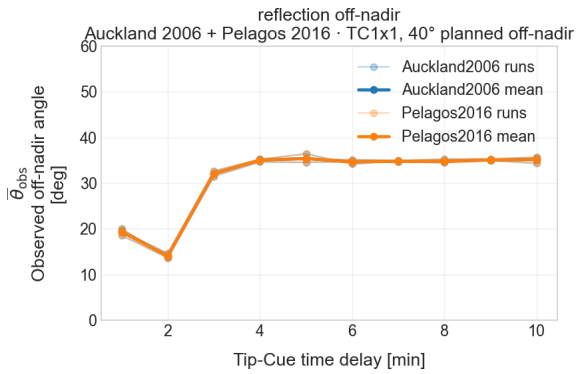


Figure 7.9: Mean observed off-nadir angle as a function of Tip-Cue time delay for the TC1x1 configuration with a maximum allowed off-nadir angle of $40°$, evaluated on Auckland2006 and Pelagos2016. Individual runs and mean are shown.

At short time delays, fewer tasks are handled, but they are observed at a lower off-nadir angle, down to approximately $15°$ instead of the allowed maximum of $40°$. This occurs because small time delays do not always allow the Cue satellite to jump sufficiently far ahead and point toward the target. At delays below approximately 2 minutes, thus below the Tip processing and transmission time of $90 + 10$ seconds plus Cue pointing time, backward pointing can even be required. As a result, the targets are observed at smaller off-nadir angles than allowed. With fewer observations, but under better viewing geometry, the number of successful whale detections remains approximately stable.

Response Metrics At time delays below the Tip processing and transmission time plus Cue pointing time, the maximum allowed off-nadir angle is not always reached. This also affects the latency, as shown in Figure 7.10. At smaller Tip-Cue delays, the Cue satellite has to point backward. The minimum latency occurs when the Cue satellite reaches the target at nadir, which happens when the time delay between Tip and Cue is approximately equal to the Tip processing and transmission time.

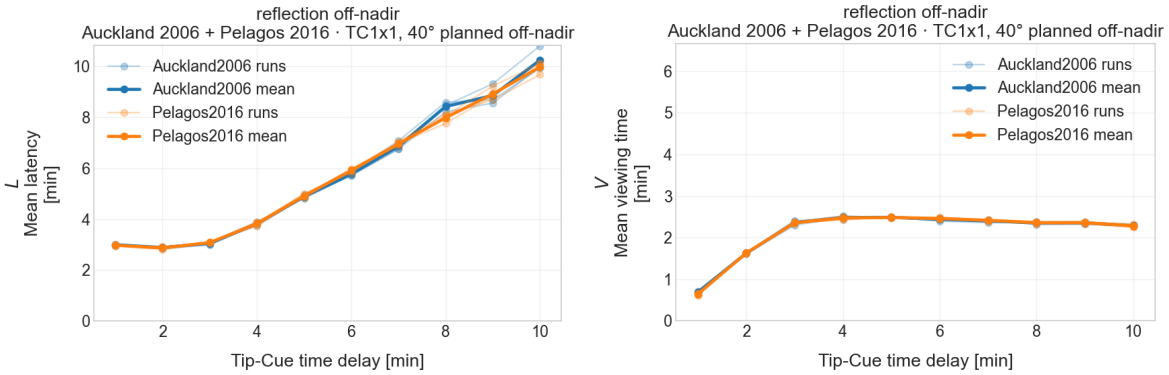


Figure 7.10: Response metrics as a function of Tip-Cue time delay for the TC1x1 configuration with a maximum allowed off-nadir angle of $40°$, evaluated on Auckland2006 and Pelagos2016. Left: mean latency of successful observations, L. Right: mean viewing time of successful observations, V.

Although shorter Tip-Cue delays reduce both off-nadir angle and latency, they also reduce the available viewing time per target. At later acquisition, there is less forward pointing capability and the target can be followed for a shorter time. This can become disadvantageous when the number of targets increases and more flexible task planning is required.

Detection Metrics Because the observed off-nadir angle remains approximately constant over most of the delay range, except for very short delays below about 3 minutes, the detector metrics are also expected to remain approximately constant. The results are shown in Figure 7.11.

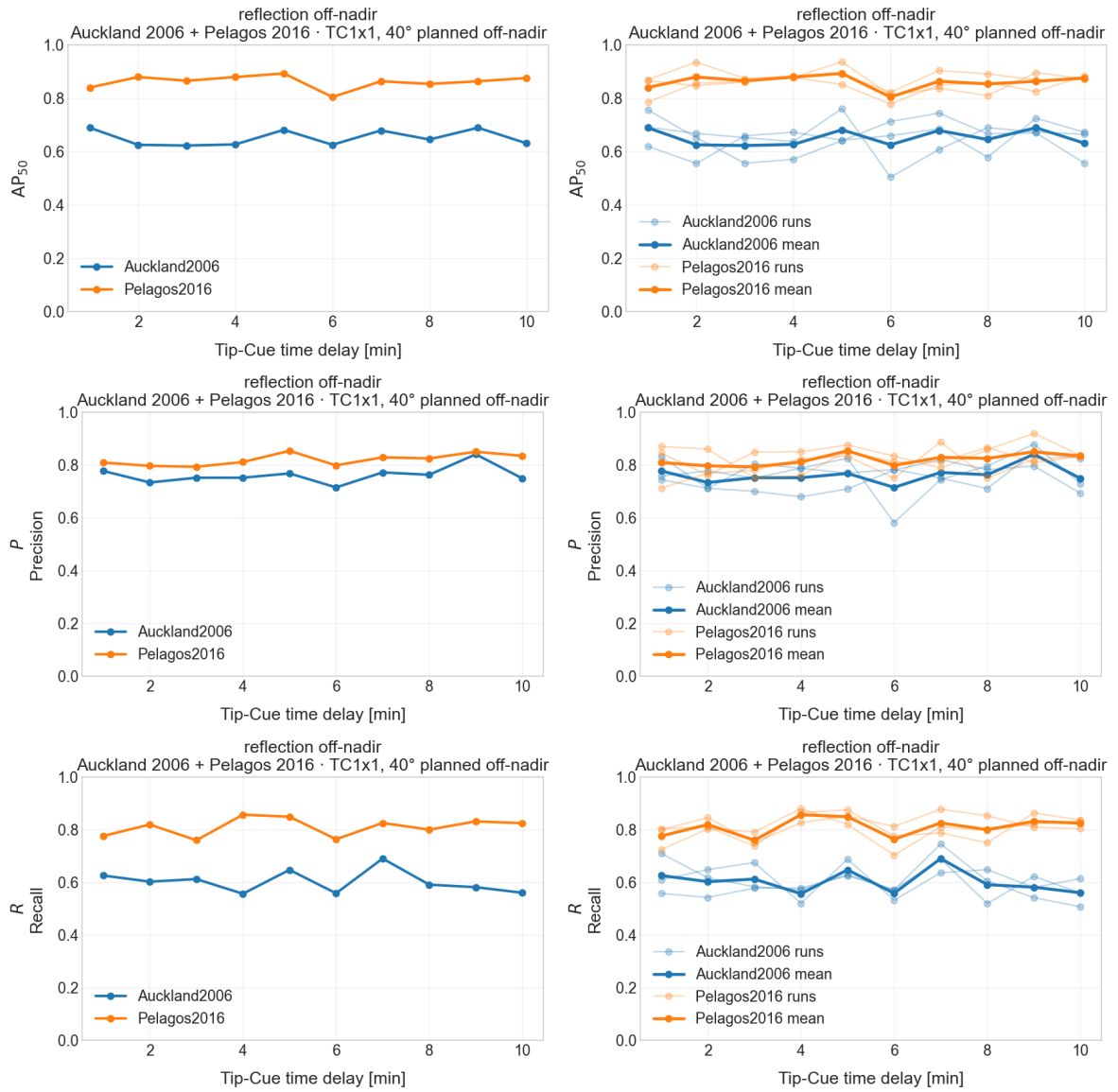


Figure 7.11: Detection metrics as a function of Tip-Cue time delay for the TC1x1 configuration with a maximum allowed off-nadir angle of 40°, evaluated on Auckland2006 and Pelagos2016. From top to bottom: AP_{50}, precision, and recall. Left: mean across runs. Right: individual runs and mean.

The local drop around 6 minutes is likely caused by an outlier, where one run shows a steep precision decrease. Likewise, recall increases again at 5 and 7 minutes, which makes this effect more pronounced. This may be caused by randomization of the sample patches, where one unfavorable set ended up in that specific run.

Overall, precision and recall remain approximately constant across the time-delay range. The successful detection capacity follows a similar trend as AP_{50} and recall, confirming that the mission-level successful detection capacity is driven both by the number of handled tasks and by detector performance, as earlier discussed in subsection 7.1.1.

Depending on mission requirements, the spacing between Tip and Cue satellite has to be optimized jointly with Tip swath width, expected target density, and latency requirements. In general, when the number of targets is low, a time delay close to the Tip processing and transmission time is favorable, because the target is then reached near nadir. However, when the number of targets increases, the additional forward pointing capability at larger spacing becomes advantageous for more flexible task planning. At minimum, the off-nadir angle has to be large enough that the full Tip field-of-view window can be reached, so that

the observation capability of both satellites is used effectively.

7.2. Imaging Effects

This section evaluates how representative Cue training data affects onboard detection performance under operational off-nadir conditions. In current literature, whale detectors are typically trained on simplified nadir-like imagery, after which they are expected to transfer to actual off-nadir observations. Here, the effect of omitting geometric effects, radiometric effects, or both is assessed, first at detector level and then within the mission benchmark.

7.2.1. Deep Learning Performance

Four models are trained. First, a full model on representative data, including both geometric and radiometric effects. Second, a model trained on data with radiometric effects only. Third, a model trained on data with geometric effects only, equal to off-nadir translation of the input patches. Fourth, a model trained on data without geometric and radiometric effects, using the original input patches only. The last setting reflects how training is still commonly done in literature. Representative examples of the four imaging configurations are shown in Figure 7.12.

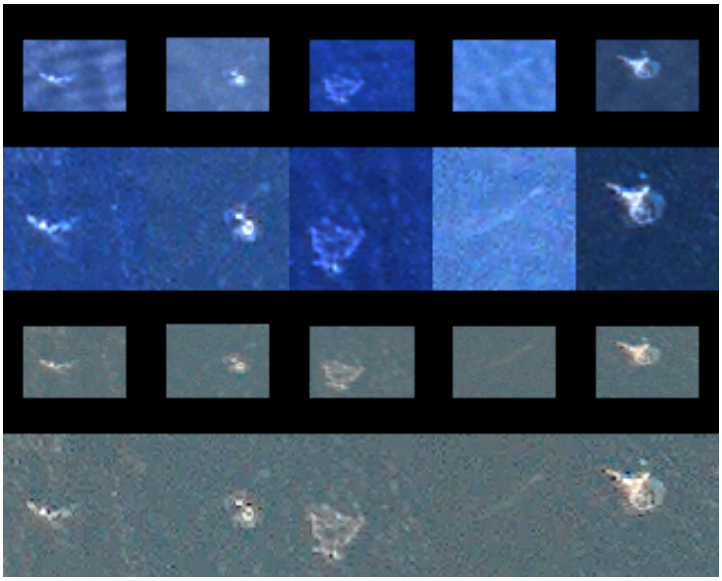


Figure 7.12: Samples of the four imaging configurations at $40°$ off-nadir following the benchmark setup. From top to bottom: both geometric and radiometric effects, radiometric effects only, geometric effects only, and no off-nadir effects. *Satellite image © 2022 Maxar Technologies.*

To further visualize the four datasets independently, examples of the full-effects, no-geometric, no-radiometric, and no-effects configurations are shown in Figure 7.13, Figure 7.14, Figure 7.15, and Figure 7.16, respectively. A more extensive overview is included in Appendix D.

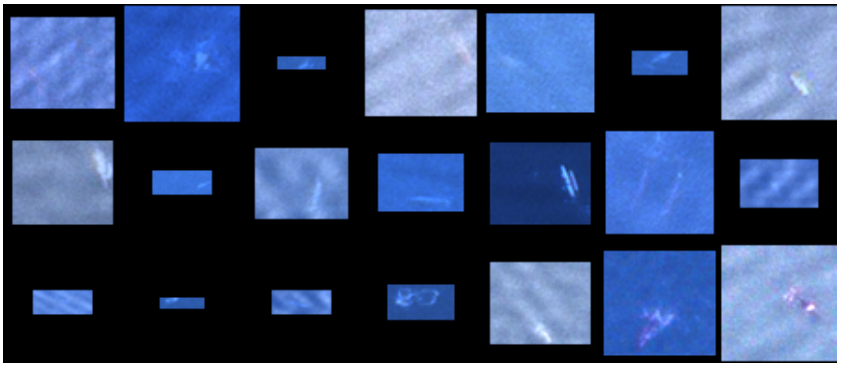


Figure 7.13: Samples of the dataset including both geometric and radiometric effects. *Satellite image © 2022 Maxar Technologies.*

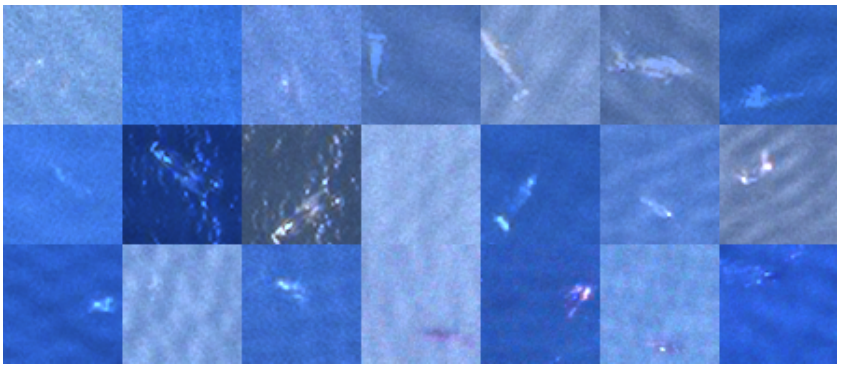


Figure 7.14: Representative samples of the dataset including radiometric effects only. *Satellite image © 2022 Maxar Technologies.*

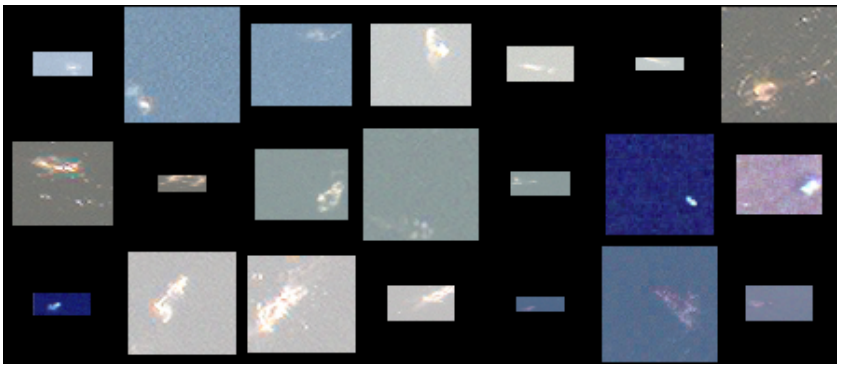


Figure 7.15: Samples of the dataset including geometric effects only. *Satellite image © 2022 Maxar Technologies.*

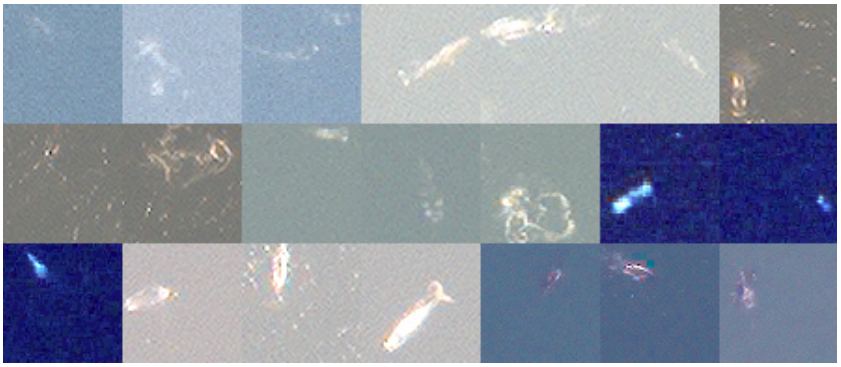


Figure 7.16: Samples of the dataset without geometric and radiometric effects, equal to the original input patches. The radiometric diversity visible across the locations shown is the only radiometric variation present in the dataset. *Satellite image © 2022 Maxar Technologies.*

First, the detector performance of the four models is evaluated by comparing AP scores and general detection performance over the full test set. To capture the overall effect, the generated test data from Pelagos2016 and Auckland2006 was combined and each model was evaluated on the full dataset. The resulting performance is summarized in Table 7.1.

Table 7.1: Performance of the four imaging configurations in the unseen-domain setting, reported separately for the validation and test sets.

Geometric effects	Radiometric effects	Validation				Test			
		AP_{50}	$AP_{50:95}$	Best F1	Thr	AP_{50}	$AP_{50:95}$	Best F1	Thr
x	x	0.894	0.470	0.962	0.380	0.757	0.343	0.930	0.296
	x	0.942	0.550	0.983	0.427	0.731	0.320	0.927	0.310
x		0.927	0.501	0.973	0.409	0.656	0.290	0.895	0.218
		0.967	0.586	0.983	0.466	0.434	0.169	0.829	0.393

The models without geometric and radiometric effects perform markedly better on the validation set. This is because their validation data follows the same simplified assumptions as the corresponding training data, and therefore matches that training domain more closely. The AP_{50} value of 0.967 is of the same order as earlier literature results on simplified whale patches, as summarized in Appendix B.

However, on the representative test data, the ranking reverses. The full-effects model performs best, with an AP_{50} of 0.757. Omitting geometric effects reduces this only slightly, to 0.731. In contrast, omitting radiometric effects reduces the AP_{50} to 0.656, a drop of approximately 0.1. Omitting both geometric and radiometric effects reduces the AP_{50} further to 0.434.

This shows that radiometric realism is the dominant factor for transfer to representative off-nadir test data, while geometric realism provides an additional but smaller gain in the aggregate metric. The full off-nadir angle dependence of these effects is evaluated later in subsection 7.1.1.

7.2.2. Benchmark Performance

After establishing the detector-level effect, the practical effect at mission level is evaluated. To do so, the four models were deployed on the Cue satellite using the confidence threshold predicted to maximize the validation F1 score, in order to balance false positives and false negatives. The true optimal test threshold is lower, but in a real operational setting that information would not be available. The performance of the four models was therefore evaluated following the benchmark defined in subsection 5.1.1.

Mission Metrics To quantify the mission-level effect, the models were deployed on the Cue satellite in the baseline scenario, with a 5 minute Tip-Cue delay and a maximum allowed off-nadir angle of 40°. The successful detection capacity was then evaluated over the off-nadir range for both Pelagos2016 and Auckland2006. The result is shown in Figure 7.17.

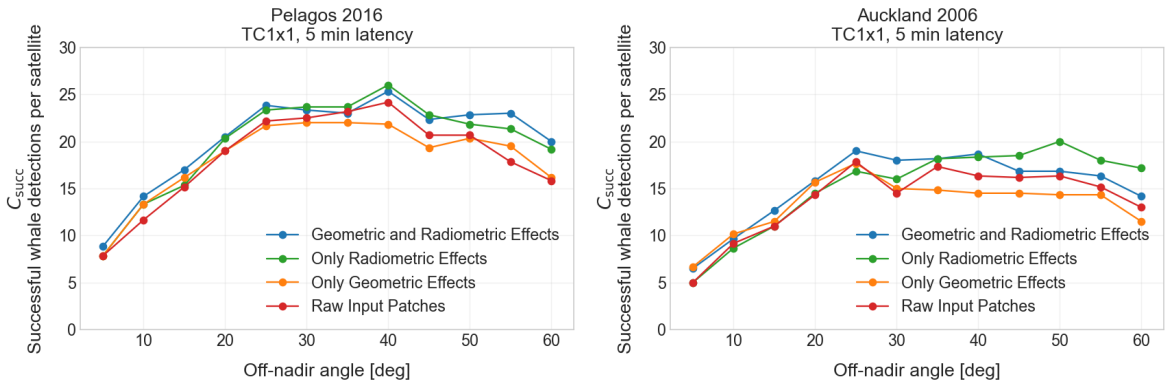


Figure 7.17: Successful detection capacity per satellite, $C_{\text{succ,satellite}}$, for the four training-data configurations over the off-nadir sweep in the baseline TC1x1 scenario. Left: Pelagos2016. Right: Auckland2006.

Overall, the models including radiometric effects perform better than the models without radiometric effects. The difference is typically in the order of 5-10 detections per satellite, corresponding to approximately 20-40% of the total performance, and becomes more pronounced at higher off-nadir angles.

In the case of Pelagos2016, the models trained with and without geometric effects perform approximately equally up to about 25°, which is expected, since both include nadir-like data. After 25°, the model trained without any off-nadir effects can temporarily outperform the model trained with geometric effects only. A likely explanation is that the raw nadir patches contain more complete image content, whereas the geometrically translated patches increasingly contain black border regions at larger off-nadir angles. As long as the whale remains visible, this can still be beneficial. From approximately 45° onwards, the models trained on nadir-like data start to degrade more strongly, and the models trained with geometric effects begin to show a clear advantage.

For Auckland2006, the overall successful detection capacity is lower for all models, which is consistent with the weaker generalization to this location discussed in subsection 6.3.4. It is also observed that the nadir-trained model can in some cases perform comparably to, or slightly better than, the model trained with geometric effects only. This was not expected beforehand, especially at higher off-nadir angles.

Detection Metrics To further assess this behavior, the detector metrics of all four models were evaluated. The response metrics were not included here, because the mission geometry remains fixed and only the detector is changed. The resulting detector performance is shown in Figure 7.18.

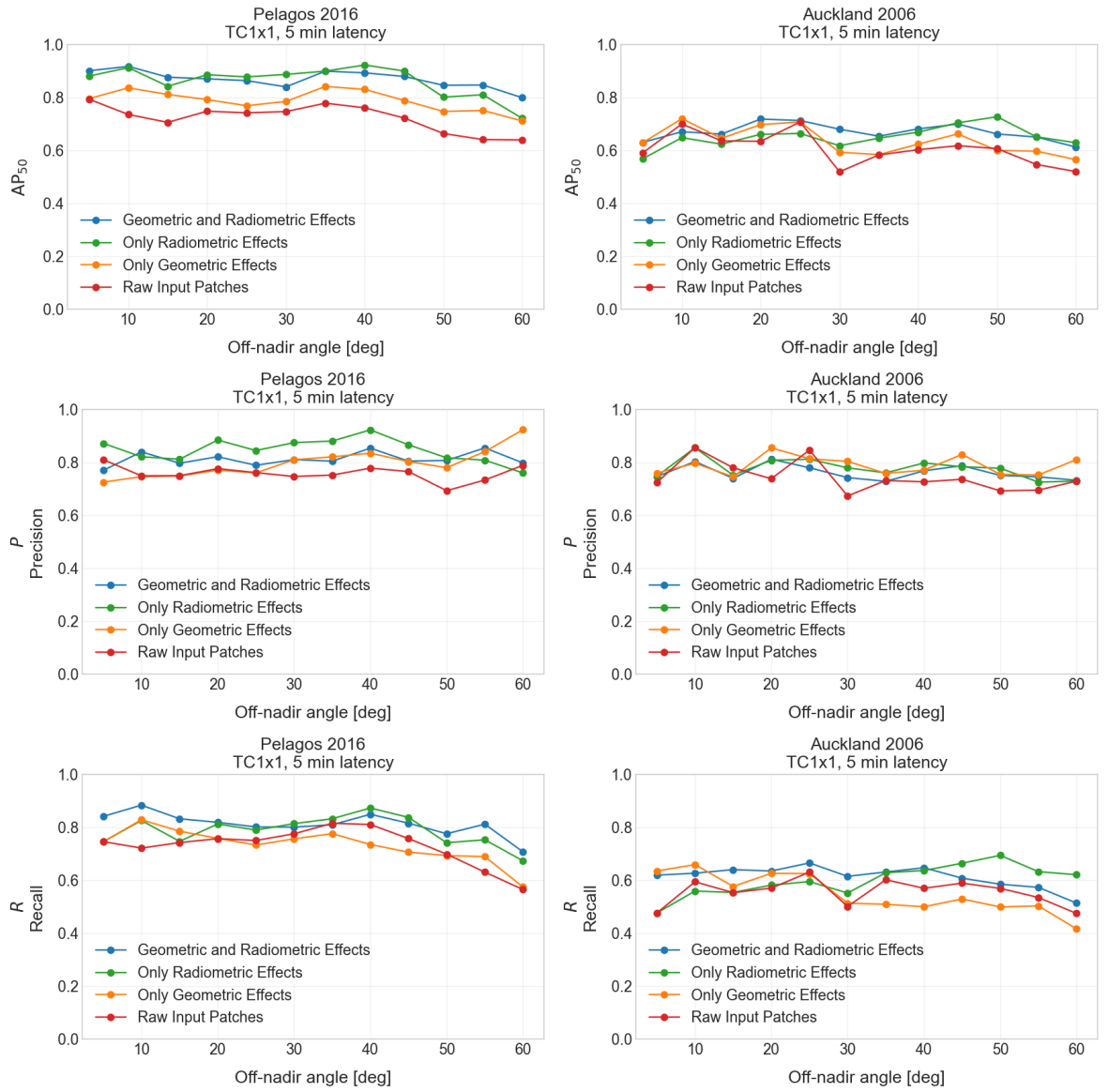


Figure 7.18: Detector performance of the four training-data configurations over the off-nadir sweep in the baseline TC1x1 scenario. From top to bottom: AP_{50}, precision, and recall. Left column: Pelagos2016. Right column: Auckland2006.

For Pelagos2016, the curves follow the expected order. Up to approximately 45°, the models trained with and without geometric effects perform similarly, while beyond 45°, the full-effects model outperforms the simplified models. This indicates that geometric realism becomes more relevant once whale appearance starts to shrink and distort more strongly.

In terms of precision, the models omitting radiometric effects can show slightly higher precision at the largest off-nadir angles. However, this goes together with lower recall. In other words, fewer false positives are made, but mainly because the model also makes fewer detections overall. The main penalty of omitting radiometric effects is therefore not a rise in false alarms, but an increase in missed whales.

The difference between including radiometric effects or not is substantial, with AP_{50} differences of around 0.1. Omitting both geometric and radiometric effects reduces AP_{50} by up to around 0.2. This indicates that Cue detector performances reported in literature on idealized whale patches are not representative of actual operational conditions.

For Auckland2006, the curves are less stable, but the same general trend remains visible. The models including radiometric effects outperform the models without radiometric effects, especially beyond

approximately 30°. Precision remains relatively similar between the models, while recall separates more clearly. This again shows that the main penalty of simplified training is an increasing number of missed whales.

The Auckland2006 recall curves also explain why the model trained on raw nadir data can in some cases match, or slightly outperform, the model trained with geometric effects only. For this location, the benefit of geometric realism appears to be partly offset by the black border regions and reduced effective image content in the translated patches. As a result, the simpler nadir patches can preserve more useful whale information for training. This effect is specific to Auckland2006 and should not be generalized.

Example detections of the default full-effects model for the baseline scenario are shown in Figure 7.19.

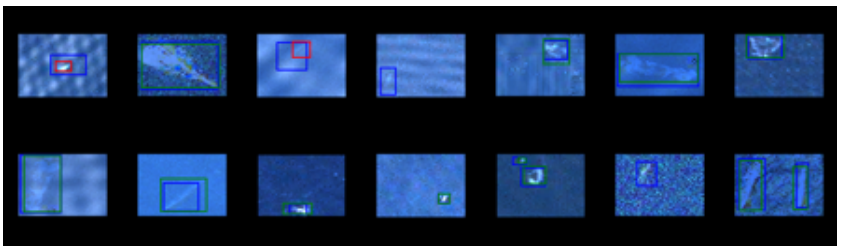


Figure 7.19: Example detections of the default full-effects model for the baseline scenario with a maximum allowed off-nadir angle of 40° and a 5 minute Tip-Cue delay.

Given the more stable generalization to Pelagos2016, this location is expected to provide the more reliable performance estimate when comparing mission configurations while keeping detector performance fixed. Therefore, the mission comparison in the next section is reported on Pelagos2016 data.

7.3. Comparative Evaluation

After analyzing mission geometry and Cue training data separately, this section combines those insights at system level. The benchmark performance of AI-Based Tip and Cue is compared with alternative satellite architectures under common reference conditions, to determine whether the coordinated concept provides a practical benefit beyond detector-level and geometry-level improvements.

All Tip-and-Cue architectures were evaluated with a maximum off-nadir angle of 40° and a Tip-Cue time delay of 5 minutes where applicable, based on the WorldView-3 orbit benchmark, using Pelagos2016 data to ensure a steady evaluation given its demonstrated generalization. No maximum observation latency was imposed. The default detector trained with both geometric and radiometric effects was used, except for the VHR constellations, where the nadir-trained detector was used because these satellites observe at nadir only.

The configurations include:

- Single Tip-and-Cue baseline: a single Tip-and-Cue system, exactly as defined in the benchmark.
- VHR satellite constellations: one constellation with 8 orbital planes and 2 satellites per plane, and one with 4 orbital planes and 4 satellites per plane. All satellites use the same swath width and spatial resolution as the benchmark Cue satellite and are built as Walker Delta constellations around the WorldView-3 orbit benchmark.
- Tip-and-Cue constellation: a Walker Delta Tip-and-Cue constellation with 4 orbital planes and 2 Tip-Cue pairs per plane, resulting in 16 satellites in total.
- Independently operating satellites: 12 satellites, consisting of 6 Tip and 6 Cue satellites. This case is based on the current actual satellite landscape: the Tip satellites are Sentinel-2A, Sentinel-2B, Sentinel-2C, and Landsat-7,8,9, while the Cue satellites are WorldView Legion 1-6. The orbital parameters were derived from these real systems, while swath width and spatial resolution were retained from the benchmark to isolate the effect of mission geometry.

It should be noted that the independently operating satellite case uses 12 satellites instead of 16, because no more than six representative Tip satellites could be derived. This also breaks the WorldView-3

benchmark geometry, since WorldView Legion 1-6 was used instead. While replacing one Cue satellite by WorldView-3 was considered, the WorldView Legion configuration better reflects the actual spread of the operational Cue satellites, and omitting one of the Legion satellites would break that geometry. Still, the benchmark timestamp was retained, and the initial positions of the independently operating satellites were derived from TLE files at that same simulation time. This ensures that the targets occur at the exact same positions across the simulations. Because the benchmark reports successful detection capacity per satellite, it remains suitable for comparison across configurations with different numbers of satellites.

The results, averaged over the three runs with different whale seeds, are shown in Table 7.2.

Table 7.2: Benchmark results for the evaluated mission architectures. Latency L is reported in minutes.

Configuration	$N_{\textbf{satellites}}$	$C_{\textbf{succ, satellite}}$	$C_{\textbf{succ, mission}}$	L **[min]**	V **[s]**	θ **[deg]**	$\textbf{AP}_{50}$	**Precision**	**Recall**
Single Tip-Cue system	2	25.3	50.7	4.9	148.9	35.4	0.893	0.853	0.849
8x2 VHR constellation	16	2.8	44.0	1.0	79.9	0.0	0.871	0.772	0.893
4x4 VHR constellation	16	2.7	43.3	1.0	79.9	0.0	0.892	0.765	0.893
4x2x2 Tip-Cue constellation	16	20.6	329.3	5.0	152.2	31.6	0.833	0.781	0.787
12 independent satellites	12	14.0	167.7	58.6	99.4	20.9	0.892	0.825	0.862

First, Table 7.2 shows that the single Tip-and-Cue system achieves the highest successful detection capacity per satellite, with 25.3 successful detections, followed by the Tip-and-Cue constellation with 20.6. This difference is remarkable, because the architecture is the same, but scaled up by a factor eight. It is likely caused by variance between the runs, which is more visible for a single configuration: the 1x1 case ranges between 23 and 28 successful Cue detections, whereas the 4x2x2 constellation ranges between 320 and 337 at mission level, as included in Appendix G. The 1x1 configuration also appeared to place the Tip satellite in a favorable position in two out of three runs. For other 1x1 cases with different latencies and off-nadir angles, the successful detection capacity typically ranged between 20 and 24. This indicates that, for a more accurate comparison, the number of targets should be increased and preferably more than ten runs should be evaluated. With more satellites, the variance between runs also decreases, making the constellation-level results more steady.

Considering the VHR constellations, the successful detection capacity is nearly independent of orbital configuration, with 8x2 and 4x4 giving almost identical results. This is expected, because both cases use the same number of satellites and cover the same simulation duration. However, both VHR constellations severely underperform compared to the Tip-and-Cue systems. Despite using eight times more satellites than the single Tip-and-Cue baseline, the total number of successful detections is still lower, with 43-44 confirmed detections compared to 50.7 for the single Tip-and-Cue system. The successful detection capacity per satellite is around 2.7-2.8 for the VHR constellations, compared to 20-25.3 for the Tip-and-Cue systems. The only clear advantage of the VHR constellations is latency, where nadir observation leads to approximately 1 minute mean delay and $\theta = 0°$.

The independently operating satellite case, which reflects how Tip and Cue has typically been studied before rather than in formation, also underperforms compared to the Tip-and-Cue formation. It reaches 14.0 successful detections per satellite, compared to 20.6 for the Tip-and-Cue constellation, while the latency increases from 5.0 minutes to 58.6 minutes. At mission level, the Tip-and-Cue constellation achieves 329.3 confirmed detections compared to 167.7 for the independently operating case, but this difference should be interpreted together with the larger number of satellites in the constellation case. The main result is therefore that the coordinated Tip-and-Cue formation achieves both higher successful detection capacity per satellite and significantly lower response latency.

To observe where this gap occurs for the independently operating case, Table 7.3 shows the task-flow metrics across the different stages.

Table 7.3: Benchmark task-flow results for the evaluated mission architectures. The first four columns are averaged from the raw `Overview` sheets across the available runs. The final column is equal to the previously reported mission-level successful detection count $C_{\text{succ, mission}}$.

Configuration	Tasks, Tip observed	Tasks, Tip sent	Tasks, Cue started	Tasks, Cue observed	Tasks, Cue confirmed, GT
Single Tip-Cue system	71.3	64.0	64.0	64.7	50.7
8x2 VHR constellation	0.0	0.0	0.0	53.0	44.0
4x4 VHR constellation	0.0	0.0	0.0	54.0	43.3
4x2x2 Tip-Cue constellation	578.0	466.7	465.3	469.0	329.3
12 independent satellites	385.3	294.7	578.3	223.3	167.7

First, the number of Tip targets detected matches the expected ratio, where $385.3/578$ approximately matches the $3/4$ ratio of 12 with respect to 16 satellites in the Tip-and-Cue constellation. The same holds for the number of tasks sent. Then, notably, 578.3 Cue tasks are started in the independent case, which is substantially more than the number of tasks sent by Tip.

This occurs because, at high latency, the Cue satellite starts pursuing a target after it has already moved out of view. When the task queue is not cleared after that failed attempt, the same target can be selected again, even though it is no longer observable. The Cue satellite then starts the same task again and fails again. As a result, tasks are repeatedly started without leading to successful observations.

Tracing the logs confirms this. In run target seed 1, this first occurs when the Cue was tasked at time step 3680, corresponding to simulation time $1{:}01$, while the target had been tipped at time step 2380, corresponding to $0{:}39$. After an observation delay of 22 minutes, the target had moved out of the field of view. With an average whale velocity of approximately 5.4 km/h, a displacement of around 2 km is sufficient to make the whale disappear from the 13.1 km Cue swath. This confirms that the missed observations in the independently operating case are caused by moving targets combined with long latency.

Given the relatively small number of targets modeled, this wasted occupancy is not expected to have dominated the final performance. Still, after this observation, the framework was updated to clear a target from the queue after a failed observation attempt when the target had moved out of view, consistent with the forward-follow-up capability of the AI-Based Tip-and-Cue framework.

While moving targets and observation latency did not appear critical earlier, because no latency requirement had been imposed, they become crucial in the independently operating case. This also shows why modelling moving targets turned out to be important for the final framework. Although whale submersion was not modeled in the end, it would be expected to reduce the successful detection capacity of the independently operating case even further, since after nearly one hour of latency, most whales would no longer remain observable at the surface.

The full results across simulation runs, for these comparative experiments as well as for the latency and off-nadir sweeps, are available in Appendix G.

8

Discussion

This chapter interprets the results in relation to the research question and the broader Earth Observation context. First, section 8.1 summarizes the main findings and relates them to earlier work. Next, section 8.2 discusses the implications for mission design and the significance of the developed framework. Finally, section 8.3 and section 8.4 address the main assumptions, limitations, and directions for future work.

8.1. Results Analysis

8.1.1. Summary of Key Findings

Following the results, orbital configuration and off-nadir imaging geometry jointly determine AI-Based Tip and Cue mission performance.

The off-nadir angle determines latency, viewing time, reachability, and detection performance, and with that the overall successful detection capacity. In general, increasing off-nadir angle lowers latency through forward pointing, increases reachability through sideward pointing, and increases viewing time through both forward and backward pointing. This comes at the cost of reduced ground sampling distance, and therefore reduced detection performance.

Up to $25°$, successful detection capacity increased because more targets within the Tip window could be reached, latency was reduced, and viewing time increased. Beyond that point, successful detection capacity was mainly driven by detection performance, which is strongly affected by the ground sampling distance of the sensor. For Pelagos2016 data at 0.31 m ground sampling distance, detection performance remained stable up to $40°$, after which it gradually dropped. For Auckland2006 data at 0.45-0.54 m ground sampling distance, this drop shifted to lower off-nadir angles. This means that optimal detection capacity occurred at the reachability ceiling, where all Tip-detected targets could be reached, but below the off-nadir performance limit.

The time delay between Tip and Cue satellite also affects latency, viewing time, reachability, and the final off-nadir angle at which the target is observed, and thereby detection performance. At larger time delays, the Cue satellite can point further forward, which increases reachability and viewing time. At smaller time delays, the Cue approaches the target earlier, such that the maximum off-nadir angle is typically not reached because maximum slew cannot be achieved within the available time. This reduces reachability and the number of targets reached, but shifts observations towards more favorable lower off-nadir angles and lowers latency. As a result, shorter delays improved observation geometry but reduced accessibility, whereas larger delays increased accessibility but shifted observations to higher off-nadir angles, causing successful detection capacity to remain relatively stable over the tested delay range.

Overall mission architecture also significantly affects AI-Based Tip and Cue system performance, with the choice between formation-based Tip and Cue or utilizing existing independently operating satellites. A coordinated Tip and Cue formation reduces latency and improves Tip-tasked target reachability. It therefore outperformed independently operating Tip and Cue satellites based on the actual Sentinel-2, Landsat, and WorldView Legion configurations. In that case, approximately one third of detected Tip

targets had moved out of view upon Cue detection or were not covered by the Cue legion. AI-Based Tip and Cue performance was also compared to traditional VHR constellations. Comparable performance to the 16-VHR satellite system, which lacks a wide-swath search stage, was achieved with only 2 satellites, corresponding to an 87.5% reduction in satellite count.

From a mission evaluation perspective, representative training data and simulation dynamics turned out to be important for reliably quantifying mission performance. Because Cue imaging relies on off-nadir pointing, both geometric and radiometric conditions have to be accounted for. Models trained on the original input patches, lacking geometric and radiometric representation, overestimated operational Cue performance by up to 0.2 AP_{50} at the off-nadir limits on the test set. Including radiometric effects turned out to be more important than including geometric effects, leading to a drop on the order of 0.1 AP_{50} along most of the off-nadir angle range.

In addition, target motion turned out to be an important factor when the latency between Tip and Cue observation increases. For Tip and Cue formation, whale motion did not dominate final performance given the rapid follow-up. For independently operating Tip and Cue satellites, however, the same target motion became a major source of missed confirmations, mainly driven by Cue window swath and target motion speed.

8.1.2. Interpretation of Results

Based on these results, AI-Based Tip and Cue mission performance should be interpreted as a trade-off between detection quality, viewing time, and latency. Reachability remains an important part of this interaction, mainly because it determines whether gains in viewing geometry and timing can actually translate into successful confirmations.

The off-nadir results imply that increasing the allowable Cue off-nadir angle is beneficial only as long as the gain in reachability, reduced latency, and increased viewing time is not outweighed by the loss in detection performance. In that sense, the optimal detection capacity is achieved when all targets within the Tip window can be reached, while the allowable off-nadir angle remains below its detection limit. The stronger effect of radiometric realism and the retained performance on Pelagos2016 data with larger whales further indicate that the models generally tolerate changes in ground sampling distance up to the point where whales vanish into the background, after which inclusion of representative off-nadir data becomes important again.

The time-delay results imply that Tip-Cue spacing, despite producing relatively stable successful detection capacity, remains an important design parameter. Shorter delays reduce latency and shift observations towards lower off-nadir angles, but at the same time reduce reachability and the number of targets reached. Larger delays increase reachability and viewing time through forward pointing, but shift observations towards higher off-nadir angles. The relatively stable successful detection capacity over the tested delay range therefore indicates that these effects compensate each other over a broad range.

The architecture comparison implies that the main benefit of AI-Based Tip and Cue lies beyond adding onboard AI to existing observation chains, and that AI-Based Tip and Cue formations should instead be assessed as a coordinated mission concept. The reduced performance of the independently operating case shows that long delay limits confirmation feasibility for moving targets. The comparison with the VHR constellations further shows that high-resolution capacity is not sufficient when the system lacks a wide-swath search stage. Within the present benchmark, the strongest performance therefore came from coordinated Tip-and-Cue formations.

The training-data results imply that mission performance cannot be evaluated reliably from simplified nadir-like data, as is common in most traditional approaches. Off-nadir geometry affects the geometric and radiometric image conditions under which the target is observed. Under those conditions, non-representative training and evaluation data directly lead to non-representative mission conclusions.

Taken together, these results show that orbital configuration and off-nadir imaging geometry influence AI-Based Tip and Cue systems through their combined effect on detection quality, viewing time, and latency. The most effective mission designs are therefore obtained by balancing these factors at system level.

8.1.3. Context and Comparison

The literature reviewed earlier treated onboard AI and autonomous tasking for Tip and Cue largely as separate topics, while neglecting off-nadir effects. As a result, earlier work did not allow direct evaluation of how orbital geometry changes Cue image conditions and how those conditions affect detection and ultimate mission performance. In that context, this work contributes an end-to-end quantification of AI-Based Tip and Cue performance by coupling orbital visibility, off-nadir image generation, and onboard detection in one evaluation chain. The results show that these components cannot be assessed independently when the final objective is mission evaluation, and enable to quantify AI-based Tip and Cue performance with respect to traditional mission architectures.

With that, the results extend earlier Tip-and-Cue studies by showing that AI-Based Tip and Cue goes beyond simply adding onboard AI to an existing observation chain, but shows strongest potential in formation geometries. Traditional work considers independently operating Tip and Cue satellites, but the increased latency causes a substantial loss of confirmation opportunities. AI-Based Tip and Cue should therefore be considered as a coordinated mission architecture in which Tip and Cue formations directly collaborate. The comparison with the VHR constellations further backends this, showing that AI-based Tip and Cue formations can achieve a similar detection capacity with a substantial lower number of satellites.

More broadly, the results show that geometric and radiometric effects cannot be neglected in off-nadir onboard AI detection chains, which have not been considered in prior work. Under those conditions, non-representative training and benchmark data translate directly into non-representative mission conclusions.

8.2. Research Implications and Significance

This thesis provides an end-to-end framework for AI-Based Tip and Cue mission analysis by coupling orbital simulation, off-nadir image generation, and onboard detection in one workflow. This is a direct contribution to the limitedly explored field, because it enables AI-Based Tip and Cue performance to be quantified as a complete mission concept rather than through separate subsystem analyses. Thereby, it allows its value to be assessed against traditional Earth Observation architectures for practical Earth-monitoring applications.

From a methodological side, it demonstrated that AI-Based Tip and Cue systems should be benchmarked in terms of end-to-end successful detections under representative observation conditions. For AI-Based Tip and Cue, mission geometry determines both whether a target can be reached and under which off-nadir and radiometric conditions it is observed. Those conditions in turn affect detector performance and therefore the final number of successful confirmations. Evaluating orbit design, imaging, and detection separately therefore gives only partial mission insight when the objective is to design Earth Observation missions that can reliably detect and confirm relevant targets on Earth.

A second contribution is the dataset-generation framework itself. The off-nadir rendering pipeline provides a physically grounded way to generate mission-relevant training and evaluation data from small annotated VHR datasets. Because the annotations are resampled through the viewing geometry and combined with radiometric variation, the generated data go beyond simple image augmentation by translation, rotation, or brightness adjustment. This is particularly relevant for data-scarce Earth Observation applications, where acquiring large labelled datasets is often not feasible.

The framework is also relevant beyond the whale use case. Since the generated data are tied to orbital geometry and sensor settings, the same workflow can be used to prepare datasets for other missions, sensors, and target classes. This makes the framework usable for both onboard detection development and for mission design, and positions it as a general tool for evaluating how sensing geometry propagates into final AI-based mission performance.

8.2.1. Mission Design Recommendations

Based on the above findings, several mission design recommendations can be derived. The results indicate that AI-Based Tip and Cue mission design should be treated as a trade-off between off-nadir angle, latency, target reachability, and available viewing time. Reducing Tip-to-Cue delay can improve the follow-up geometry and shorten confirmation time, but it can also reduce the available observation window. Increasing the allowable off-nadir angle improves reachability, but also increases geometric distortion and radiometric degradation.

A first recommendation is that, for high-performance applications with tight detection time limits, AI-Based Tip and Cue formations should be preferred over independently operating Tip and Cue architectures. The use of existing independently operating satellites may instead be preferred when financial and mission design budgets are limited. VHR constellations should only be preferred when insufficient spatial detail can be extracted from the wide-area search stage for useful tipping.

When moving forward with AI-Based Tip and Cue formations, a second recommendation is to match the maximum allowable Cue off-nadir angle to the effective width of the Tip observation window. In a formation-flying concept, there is little benefit in designing a Tip window that extends beyond the area the Cue satellite can actually reach. When the Tip swath width and Cue pointing capability are selected together, the full detection window remains operationally usable.

In addition, after obtaining a maximum off-nadir angle limit, the time delay between Tip and Cue satellites should be set based on this limit to achieve minimum latency. This is achieved when the Cue satellite reaches its maximum off-nadir pointing exactly after the Tip processing, transmission, and Cue slewing time have passed. The corresponding geometrical spacing should therefore be added to these delays to determine the total spacing between Tip and Cue.

Regarding Tip-Cue tasking, a general recommendation is to improve tasking logic through imaging-condition awareness and target relevance. Concerning imaging-condition awareness, the current framework already links mission geometry to imaging quality, which means that future tasking cost functions could account for expected off-nadir angle and sunglint conditions in addition to target reachability when prioritizing targets. This would allow Cue scheduling to favor observation opportunities that are more likely to result in usable detections.

Targets should also be prioritized based on their relevance. In operational missions, the information available at the Tip stage can be used to rank targets before Cueing, for example based on target characteristics, confidence, urgency, or operational interest. This would make the framework more suitable for missions in which limited Cue opportunities need to be allocated to the most valuable targets first.

Based on the above findings, an AI-Based Tip and Cue mission design procedure is presented below.

1. *Application selection.* The mission design is driven by the selected application. AI-Based Tip and Cue is most relevant when target location is uncertain, rapid follow-up is required, and confirmation requires higher spatial detail than the initial search.
2. *Mission requirements.* The application shall be translated into quantitative requirements, including revisit time, acceptable confirmation latency, approximate spatial resolution, evaluation properties, and required confirmation reliability.
3. *Architecture selection.* Based on these requirements, the overall mission architecture shall be selected, either as a Tip and Cue formation when latency is critical, as independent Tip and Cue to utilize existing satellites, or as a VHR constellation when the tipping stage would otherwise not allow sufficiently useful observations because of limited spatial resolution.
4. *Preliminary sensor definition.* In case of a choice for Tip and Cue, preliminary Tip and Cue sensor specifications shall be selected. The Tip sensor should provide sufficient swath width and search capability, while the Cue sensor should provide sufficient spatial detail and image quality for reliable target confirmation.
5. *Initial detector training over the candidate off-nadir range.* Before mission-level design is finalized, the onboard Cue detector should first be trained and evaluated over the full candidate off-nadir range under representative operational conditions. To do so, the simulation can be run with an off-nadir angle range and a provisional orbit, for example derived from comparable Cue satellites, to obtain geometrical input parameters for the dataset. This step provides an initial estimate of how detection performance changes with viewing angle.
6. *Initial Cue performance over off-nadir angle.* Based on this initial detector training, Cue confirmation performance should be quantified as a function of off-nadir angle. This is required because Cue observations are generally acquired away from nadir, and confirmation performance degrades with increasing viewing angle.

7. *Provisional operational Cue off-nadir range.* From this initial performance characterization, a provisional maximum allowable Cue off-nadir angle should be defined. This limit serves as an initial system design input.
8. *Tip window matched to Cue reach.* The Tip observation window should then be matched to the area that remains reachable by the Cue segment within the allowable off-nadir range. In a coordinated system, there is little benefit in searching areas that cannot subsequently be exploited for confirmation.
9. *Tip-Cue spacing and timing.* The relative spacing between Tip and Cue should be selected such that, after Tip detection, processing, task transmission, and Cue slewing, the Cue satellite can observe the target within the limits of the allowable off-nadir range. A placement close to this limit maximizes viewing time and targeting flexibility, whereas lower off-nadir placements favor better detection conditions.
10. *Orbital configuration and constellation sizing.* Once Tip swath, Cue off-nadir range, and Tip-Cue spacing are approximately consistent, the orbital configuration should be selected. This includes the orbital geometry, planes, and number of satellites required to satisfy coverage and revisit requirements.
11. *Tasking logic.* Tasking should then be customized so that follow-up opportunities are prioritized based on Tip satellite data and expected observation conditions that drive image quality. In that way, relevant observations with more favorable viewing conditions can be preferred.
12. *End-to-end evaluation.* The complete system should then be evaluated at mission level by combining orbital geometry, imaging conditions, and detection performance. Accessibility, latency, viewing time, and successful confirmations should be assessed following the benchmark.
13. *Detector refinement around the operational angle range.* Based on the end-to-end simulation results, run with different target seeds until a sufficient range of observation conditions has been collected, geometrical target observation parameters can be extracted. From these, the actual distribution of Cue observation angles can be identified and used as input for a mission-specific training dataset. The detector can then be retrained or fine-tuned using data concentrated around this operational range, and where relevant slightly beyond it, in order to improve confirmation performance in the relevant part of the viewing envelope while retaining generalization.
14. *Updated Cue off-nadir range.* After detector refinement, the operationally acceptable maximum Cue off-nadir angle should be reevaluated by deploying the retrained model over the expected off-nadir-angle data range. This step can shift the optimum off-nadir limit, because improved detector robustness in the mission-relevant angle regime may justify a different trade-off between detection quality, latency, and viewing time.
15. *Design iteration.* Based on that, Tip swath, Tip-Cue spacing, orbital configuration, detector training range, and maximum off-nadir angle are revised within the same loop. The final design should be obtained from an iterative process investigating the coupling between them. In the last step, with additional training data from the specific configuration and a broader range retained for generalization, the detector can be retrained to maximize its operational performance before deployment.
16. *Final design point.* The selected mission design should be the configuration that best satisfies the application requirements at system level. Compliance with the system requirements is verified by running the end-to-end simulation for different target distributions, initialized with different seeds and propagated under representative target dynamics.

8.2.2. Framework Extendability

The framework is directly extendable beyond whale detection and Tip and Cue. Because the dataset-generation method can be rerun under different simulation settings, it can be used to expand training data from limited VHR input data at a large scale. Within the present setup, 633 input patches were expanded to 13,925 generated images. This is particularly valuable for data-scarce Earth Observation applications, where labelled training data are limited.

Because the generated data are tied to orbital geometry and sensor settings, the framework can also be used during mission design to prepare datasets for a specific orbit or sensor configuration. After

simulating the relevant mission conditions and passing these into the off-nadir imaging pipeline, training data can be generated for the expected acquisition geometry and radiometric conditions of the mission. This makes it possible to train detection models for the sensing conditions the satellite will face already before launch. With multispectral inputs, the same approach can also be extended beyond RGB to mission-specific spectral bands.

The framework also extends beyond the whale use case and beyond AI-Based Tip and Cue alone. The orbital simulation can be reused independently for other Earth Observation missions and other target types by changing the orbital and sensor design inputs and recomputing the resulting observation opportunities and viewing geometries. This makes it possible to evaluate how mission design choices, such as field of view, maximum off-nadir angle, orbital geometry, and constellation definition, affect expected detection performance before deployment.

Because the workflow couples geometry, rendering, and detection, it is also suitable for systematic sensitivity analysis. This is useful both scientifically and practically. Scientifically, it allows quantification of which mission variables drive performance most strongly. Practically, it allows mission designers to identify whether a performance bottleneck lies mainly in orbit design, in sensing conditions, or in the detector itself.

8.3. Assumptions and Limitations

8.3.1. Orbital Simulation

The orbital simulation is built for mission-level comparison, but several assumptions limit how directly the results can be transferred to a specific operational case. Onboard processing and communication delays are represented by fixed values, and the Cue segment is evaluated using a WorldView-3-like reference configuration. This gives a consistent benchmark across all simulated cases, but it also means that the reported latency and tasking results are tied to that mission setup. For use in a specific mission design, these parameters would need to be replaced by mission-specific values.

The target propagation is also tailored to the whale detection use case. Whale motion is included in simplified form, and within the current configuration it did not appear to be a dominant driver of the final performance in the coordinated Tip-and-Cue formations. However, this should not be assumed to hold more generally. For faster or less predictable targets, vessels for example, more detailed target dynamics and uncertainty propagation would be required. In the same context, targets are assumed to remain visible once present. Whale submersion is not modelled, which likely makes the confirmation results optimistic at larger Tip-to-Cue delays, since longer delays increase the chance that the target submerged when the Cue satellite arrives. Including resurfacing and temporary invisibility would therefore make the latency sensitivity more realistic.

The tasking logic is implemented as a local and instantaneous decision rule. Each Cue satellite selects, from the currently feasible options, the target that can be reached first. This captures the effects of line of sight, off-nadir accessibility, and slew time on task completion, but it does not optimize the full observation sequence. In practice, tasking should also account for the complete target list, including target priority, target certainty, visibility windows, spatial clustering, and the effect of the current decision on later observation opportunities. Replanning during a maneuver would also be important, especially when new detections enter the queue or when a more relevant target cluster appears. This was not critical in the present study because the benchmark uses 500 homogeneous targets, but it would become important in a more operational scenario where mission utility, rather than earliest reachability, drives the planning.

Finally, the pointing and stabilization model is kept at system level. The current implementation computes the required target attitude and applies a simplified slew and stabilization response based on linear dynamics, which is sufficient to evaluate whether a target can be reached and imaged within the required time. However, it does not represent full spacecraft attitude control behavior. Specifically, the current Euler-angle-based path does not include a higher-fidelity control formulation or mission-specific actuator dynamics. More advanced control approaches, such as PID- or MPC-based schemes, were explored and implemented, but did not yet result in a robust Euler-angle propagation within the scope of this work. For detailed mission planning, this part of the framework should therefore be replaced by a higher-fidelity ADCS model that better reflects the actual spacecraft dynamics.

8.3.2. Off-Nadir Imaging

The off-nadir imaging pipeline captures the main geometric and radiometric effects needed for this study, but several assumptions still limit its radiometric realism. Most importantly, the outgoing atmospheric path is not modelled. The Mitsuba sensor captures the radiation reflected by the texture directly, without accounting for absorption and scattering between the surface and the satellite. As a result, the rendered output is closer to a bottom-of-atmosphere signal than to a true top-of-atmosphere satellite measurement. Including atmospheric effects in the outgoing ray direction would therefore be required if higher radiometric accuracy is desired.

The ocean surface itself is also represented through an estimated wave DEM. This is sufficient to introduce non-flat surface structure and directional reflection effects, but it remains an approximation of the actual sea state. The same holds for the water reflectance modelling. Two BSDF formulations are implemented, but both remain estimates of the true surface radiative behavior.

The texture input introduces an additional limitation. The ocean patches originate from different satellite images with different GSDs and different levels of post-processing. In the current pipeline, these inputs are used as textures and are then observed by one predefined simulated sensor, which avoids the need to model each source sensor individually. However, the texture content still inherits radiometric and visual characteristics from the original imagery. This is especially relevant because the input RGB textures are gamma-corrected and were not preprocessed before entering the rendering pipeline. For most cases this did not create a major issue, but it was noticeable for Valdes2016, where the lower input reflectance led to stronger scaling effects. A more consistent preprocessing step could further reduce the dependence on the original input image statistics. Since the current RGB textures are gamma-corrected, an improvement would be to first reverse this gamma encoding to recover a more linear signal before rendering, closer to the actual scale of radiance and reflectance. In addition, a separate brightness normalization step could be applied to make the input textures more consistent across datasets and to reduce differences caused by the source imagery.

A further assumption is introduced when the whale mask is applied. Reflection scaling is derived from the surrounding ocean pixels, which assumes that the whale follows the same reflection behavior as the nearby water. This choice was made to avoid introducing artificial model bias through augmentation-specific whale radiometry. At the same time, it also means that the resulting images are less physically representative of the true whale-water contrast.

Regarding the validation chain based on comparison with the Φ-Sat-2 data, the satellite spectral response curve is estimated because the actual instrument data were unavailable. Any mismatch in the shape or integral of this curve over wavelength affects the amount of received radiation and therefore the final image brightness. As a result, a direct radiance-level comparison was not possible. In addition, the surface wind speed had to be estimated, since it is not available in the input imagery.

8.3.3. Onboard Detection

Besides the limitations inherited from the rendering pipeline, the onboard detection stage introduces several additional assumptions. First, the dataset was intended to be approximately class-balanced, with a similar number of whale and ocean patches. In practice, this was not always achieved. In some cases, an ocean-only patch could not be cropped because the whale occupied too much of the available area. In addition, patch generation failed more often at higher off-nadir angles, for example when projected bounding-box corners mapped very close to each other. These cases were treated as failed detections, since such projections were assumed to be non-detectable in practice as well. This means that patch generation failures are more frequent at higher off-nadir angles and may contribute to the observed performance reduction in that regime.

A second limitation is introduced by the black-border artifact in the generated off-nadir patches. Because the translated patches occupy a smaller central area at larger off-nadir angles, the detector effectively searches a reduced image region. This can artificially increase AP_{50} and recall while the whale remains visible, particularly in the intermediate angle range up to approximately $40°$. The corresponding increase in detector performance in that regime is therefore not fully representative of operational full-scene imagery.

A third limitation lies in the evaluation metric. IoU is not scale-aware, which becomes especially relevant at high off-nadir angles, where whale projections can shrink to only a few pixels. In that case, a one-pixel

mismatch can cause a large drop in IoU. For a 3×1 pixel target, for example, a one-pixel localization error already leads to a strong relative penalty. Lower IoU scores at larger off-nadir angles are therefore expected and do not necessarily mean that the whale was not detected. In this regime, it is more informative to consider whether the whale was detected at all than to rely only on the exact IoU value.

The detector was also evaluated on small cropped patches. The reported results therefore quantify detection performance under controlled local conditions, while performance on full-scene satellite imagery still remains to be determined. In addition, the current setup is limited to one-class whale detection, which is less demanding than distinguishing whales from a broader set of visually similar object classes. A relevant next step would therefore be to evaluate up to which off-nadir angle whales can still be reliably distinguished from other objects. That practical limit is expected to be lower than in the current one-class setup.

Regarding the validation split, Valdes2016 appeared twice in the randomly seeded cross-fold set. This was initially considered preferable for generalization assessment, but in hindsight it was not ideal, since Valdes2016 turned out not to be representative of the overall dataset. Its behavior was affected by the absence of input preprocessing, which was also reflected in the off-nadir rendering validation against the Φ-Sat-2 data in section 6.2. A more suitable split would have been one in which another location appeared twice, so that more than half of the folds provided reliable performance estimates. More generally, a larger number of cross-fold splits would have been preferable. At least seven folds, with each location appearing twice, would already have improved the robustness of the evaluation, while the full set of 21 locations would have provided the most reliable performance estimate. This was not feasible within the present study because of the available computational resources and time.

In addition, no angle-aware domain adaptation approach was evaluated. This means that the detector was trained directly on the generated off-nadir data, without using nadir imagery as supporting source data during training. [41] suggests that this can improve robustness at larger viewing angles while reducing the amount of off-nadir annotation required. Testing whether this also benefits the present detection problem would therefore be relevant future work. Additionally, the dataset could be designed more explicitly around the maximum off-nadir angle requirement, which would lead to more comparable samples during training.

Finally, the model comparison is not based on fully independent hyperparameter optimization for each setting. Hyperparameters were optimized for a representative model trained on data containing both radiometric and geometric effects, and then reused for models that isolate these effects. This was justified by the limited sensitivity to augmentation observed in the representative case. However, this transfer is not guaranteed to be optimal, where models trained on less diverse data, such as the default nadir configuration, may benefit from stronger augmentation, meaning that the selected hyperparameters may not translate one-to-one.

The same strategy was applied to the model-size comparison from Atto to X. Instead of re-optimizing each model individually, the optimized configuration was reused, with only the number of training epochs adjusted in bins to account for model capacity. This approach was preferred over the default DEIMv2 settings, as the datasets differ from the nominal benchmark domain while maintaining the same image resolution and task structure. As a result, part of the observed performance differences may be influenced by the shared hyperparameter configuration, beyond effects of data and model capacity alone. However, this was considered acceptable for an initial estimate, as augmentation settings were not found to be a decisive factor during the hyperparameter optimization.

8.4. Recommendations for Future Work

Several extensions can further improve the framework and make it more useful for mission design. First, the current detection analysis can be extended by training dedicated models for the Tip satellite. The same rendering pipeline used for the Cue data can be applied to generate Tip imagery under consistent radiometric conditions, enabling a direct comparison between Tip- and Cue-side detection performance within one unified simulation chain.

A second priority is to move from small cropped patches to larger patches or full-scene inputs. This would reduce the black-border artifacts that currently affect the off-nadir analysis, provide a more realistic performance estimate between approximately $25°$ and $40°$ off-nadir and beyond, and make better use

of the full DEIMv2 input resolution. Related to this, training directly on L0 or L1 data instead of visually processed image products would better match the intended onboard processing chain. This is particularly relevant if Tip data are to be forwarded at low processing level and Cue data are to be processed rapidly after acquisition. It should be expected that radiance-level data will also include stronger geolocation and sensor effects, but it would be more representative of an actual onboard application.

The current one-class whale detection problem should also be extended. Training and comparing multi-class detectors, for example for whales, ships, or icebergs, would make the framework more representative of operational scenarios. This is especially relevant because multi-class discrimination is more difficult than the current one-class setup and would provide a better indication of how robust off-nadir detection remains in the presence of visually competing object classes.

The current model tuning can also be made more targeted. Performance can be analyzed as a function of off-nadir angle and sunglint condition, after which additional training data can be generated specifically for the conditions under which the detector underperforms. In the same context, it would be valuable to generate a dedicated dataset around the maximum glint position to evaluate how detection degrades under strong sunglint and to fine-tune the model on these cases if needed.

Several improvements are also possible on the rendering side. The current pipeline is suitable for targeted mission-level dataset generation and off-nadir sensitivity analysis, and it provides a representative estimate of the received radiation. However, it should not be used for high-accuracy sensor calibration or tuning in its current form. Further improvement would require the use of actual satellite spectral response curves, better-constrained surface wind and wave information, preprocessing and normalization of the input texture patches, and modelling of the outgoing atmospheric path to move from a bottom-of-atmosphere-like representation toward top-of-atmosphere sensor measurements. In addition, the current RGB-based rendering can be extended to multiple spectral bands, for example including infrared channels for Tip satellite data. With these additions, the rendering pipeline could move closer to mission-specific radiometric dataset generation.

The mission simulation can also be expanded further. Since each satellite is represented as a PASEOS object, the current coupling can be extended beyond the deveoped EO and attitude module to include power usage, communication budgets, and ground-station links. This would allow the framework to evaluate also whether the corresponding data can be processed and delivered within operational constraints.

At target level, the current whale use case can be made more realistic by including submersion behavior. This would allow a more accurate evaluation of how latency affects confirmation performance when a target is only temporarily visible.

The framework can also be extended to other application domains. For marine applications, the current off-nadir rendering approach is well suited because the scene background is relatively homogeneous. Extension to land applications is also possible, for example for forest-fire detection or deforestation monitoring, but becomes more challenging when surface reflectance varies strongly within one image. Specifically, highly heterogeneous urban scenes would require more advanced reflection modelling to avoid introducing unintended model bias through the augmentation pipeline.

Finally, AI-Based Tip and Cue formations should be studied further as a mission concept. This work demonstrates their potential, but more detailed mission-level optimization is still required to maximize the performance benefit. This includes further exploration of orbital design, sensor parameters, tasking logic, onboard processing, and communication constraints. This would allow the concept to move from initial feasibility assessment toward more detailed mission implementation.

9

Applications Framework

This chapter provides an overview of potential application domains for AI-based Tip and Cue architectures, which be used to support prioritization for further development and demonstration. The applications are organized to clarify where the concept offers structural advantages and where conventional approaches remain sufficient.

The assessment is guided by five practical application considerations, which reflect the main strengths and limitations of AI-based Tip and Cue systems, as presented in Table 9.1.

Table 9.1: Architectural Evaluation Criteria for AI-Based Tip and Cue Applications.

Criterion	Description
Spatial Uncertainty	Addresses whether the target location is unknown and requires wide-area scanning before detailed observation is possible.
Spatial Detail Requirement	Considers whether high-resolution imaging is necessary for reliable decision-making.
Latency Sensitivity	Evaluates whether rapid confirmation significantly changes operational outcomes.
Spectral Sensitivity / Off-Nadir Robustness	Examines whether performance degrades substantially when observations are acquired at larger viewing angles.
Availability of Alternative Monitoring Infrastructure	Assesses whether strong ground-based, airborne, or existing space-based systems already dominate the application.

For clarity, the applications are grouped into five domains: Security and Surveillance, Ecosystem and Environment, Urban and Industry, Natural Disasters, and Space. The resulting matrix in Table 9.2 summarizes the architectural suitability across application domains.

Table 9.2: Architectural alignment of application domains with AI-Based Tip and Cue.

Alignment	Security & Surveillance	Ecosystem & Environment	Urban & Industry	Natural Disasters	Space
High Benefit	• Military operations • Illegal maritime activities • Search and rescue	• Small-scale deforestation • Wildlife conservation • Fish spotting • Rip current detection • Permanently clouded region mapping	• Methane leak detection • Oil spill detection	• Wildfire detection • Post-disaster damage assessment	• Space debris monitoring
Moderate Benefit	• Unauthorized flight detection • Illegal land activity detection	• Harmful algal blooms • Sargassum monitoring • Biodiversity monitoring • Plastic soup mapping	• Emission profile monitoring • Sewer inspection • Agriculture monitoring	• Flooding dynamics mapping • Avalanche prediction • Hurricane prediction	• Solar flare / CME monitoring • Satellite inspection • Landing site assessment
Limited Benefit	• Accident investigation	• Coral reef monitoring • Ocean dead zone monitoring • Glacier monitoring • Iceberg detection	• Power grid failure detection • Ground movement & geohazards • Traffic flow optimization • Archaeological site detection • Remote trash mapping • Solar panel inspection • Wind turbine inspection • Thermal insulation monitoring • Livestock management	• Seismic activity monitoring • Volcanic eruption monitoring • Lightning strike detection • TLE monitoring	• Aurora monitoring

9.1. High Benefit

The applications in this section show the strongest structural fit with AI-Based Tip and Cue. They combine uncertain target location, a clear value of rapid follow-up, and a need for higher spatial detail at the Cue stage.

9.1.1. Security and Surveillance

Military Operations

Military operations require rapid situational awareness of mobile assets, temporary infrastructure, and force movements [32]. Satellite imagery is widely used to identify potential threats such as ships, tanks, aircraft, missiles, and military installations, as well as to monitor troop movements and assess damage to critical infrastructure. Because these targets are often spatially uncertain and the operational value of intelligence decreases rapidly within fast decision cycles, timely observations are critical.

Monitoring these activities requires screening large areas to detect emerging activity, while high spatial resolution is needed for asset classification, intent assessment, and battle damage evaluation.

A Tip and Cue architecture can support this by enabling wide-area monitoring to detect potential activity, followed by rapid high-resolution observations for detailed identification. Reducing the time between detection and confirmation directly supports command, control, and operational planning [22], [23].

Figure 9.1: Examples of satellite imagery for military purposes, from which troop movements and target locations can be timely identified. [1] [2]

Illegal Maritime Activities: Piracy, Migration, Illegal Fishing

Illegal maritime activities including piracy, smuggling, illegal fishing, and irregular migration frequently occur in remote ocean regions where authority presence is limited and ground-based monitoring systems such as radar provide insufficient coverage [186]. In 2023, 120 incidents of maritime piracy and armed robbery against ships were reported, primarily in regions including Somalia, the Gulf of Guinea, the Singapore Straits, and the Caribbean [187]. Illegal, unreported, and unregulated (IUU) fishing also remains a major challenge. In West Africa, total catches are estimated to be about 40% higher than officially reported [3], while between 20% and 32% of wild-caught seafood imported into the United States is estimated to originate from IUU sources [188].

Detecting these activities requires monitoring large maritime areas to identify anomalous vessel behaviour, while high spatial resolution is needed for vessel classification and activity verification (Figure 9.2). Synthetic Aperture Radar (SAR) enables vessel detection under all weather and illumination conditions, whereas optical imagery provides detailed identification of vessel characteristics.

A Tip and Cue architecture can combine wide-area monitoring to detect suspicious vessel activity with rapid high-resolution imaging to verify vessel identity and behaviour. This enables large-scale surveillance while maintaining the spatial detail required for enforcement.

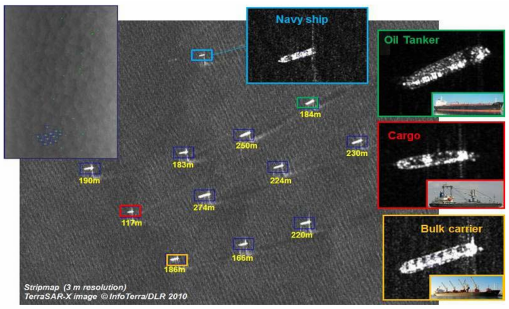


Figure 9.2: Example of SAR satellite imagery for ship detection and classification, where various types of ships can be distinguished. [3]

Search and Rescue Operations

Search and rescue operations involve locating stranded or missing individuals in remote environments such as oceans or mountainous regions. In these situations, survival probability often decreases rapidly with time, making early localization operationally critical. Currently, manned aircraft and Unmanned Air

[1] W. Swan, B., and P. McLeary, "Satellite images show new Russian military buildup near Ukraine," *Politico*, Nov. 1, 2021. Available: `https://www.politico.com/news/2021/11/01/satellite-russia-ukraine-military-518337`. Accessed: Apr. 10, 2025.

[2] S. Erwin, "Commercial spy satellites put Russia's Ukraine invasion in the public eye," *SpaceNews*, 2022. Available: `https://spacenews.com/satellite-imaging-companies-increase-profile-as-they-track-russias-invasion-of-ukraine/`. Accessed: Apr. 10, 2025.

[3] A. Zampieri, "Better early warning through detection tools: state of play of the PMAR projects," European Commission's Joint Research Centre, Brussels, Belgium, Mar. 28–29, 2012. Available: `https://transport.ec.europa.eu/document/download/48f0a8d6-01f1-4809-a2b9-d47629e96e2a_en?filename=2-2-session2-jrc-zampieri.pdf&prefLang=it`. Accessed: Apr. 10, 2025.

Vehicles (UAVs) are commonly deployed to search for missing persons in remote settings, although their effectiveness is limited by battery endurance, weather conditions, and the large spatial extent of search areas [189].

Locating missing persons requires scanning large search regions to detect possible distress indicators, while high spatial resolution is needed to identify individuals, vessels, or debris and assess their condition.

A Tip and Cue architecture can combine wide-area monitoring with rapid high-resolution follow-up observations once a potential target is detected. Higher-resolution spaceborne imagers are expected to further improve the feasibility of satellite-supported search and rescue operations in future systems.

9.1.2. Ecosystem and Environment

Small-scale Deforestation

Tropical deforestation is a major contributor to global biodiversity loss and carbon emissions, while forests cover approximately one-third of the Earth's land surface [4]. A large fraction of deforestation occurs in small patches caused by activities such as illegal logging or small-scale agricultural expansion. In the Amazon, for example, 96.4% of deforested patches are smaller than 6.25 hectares and 81.1% are below 1 hectare, yet these small clearings still account for 39% of total forest loss [190].

Monitoring deforestation therefore requires the observation of large tropical regions to detect emerging forest disturbances, while high spatial resolution is needed to confirm logging activity and associated infrastructure such as access roads or base camps (Figure 9.3). Satellite observations across visible, infrared, and SAR wavelengths already play an important role in forest monitoring. Nevertheless, detecting small-scale disturbances remains challenging due to their limited spatial extent and the difficulty of distinguishing natural canopy variation from human-induced change, which can hinder enforcement actions [191], [192].

A Tip and Cue architecture can support this task by enabling wide-area monitoring to identify potential forest disturbances, followed by targeted high-resolution imaging to confirm logging activity and associated infrastructure. Early confirmation of small clearings can support enforcement actions before deforestation expands further.

Figure 9.3: Examples of satellite imagery showcasing small scale logging activity and base camps at nadir viewing angles. [4] [5] [6]

Wildlife Conservation

Wildlife populations are declining globally due to human activities including habitat destruction, illegal wildlife trade, pollution, and climate change [193], [194]. Currently, more than 42,000 species are classified as threatened with extinction, representing approximately 28% of all species assessed by the International Union for Conservation of Nature [195]. Reliable monitoring of species distribution, population size, and migration patterns is therefore needed for biodiversity conservation and ecosystem management.

[4] S. Novoa, L. Villa, and M. Finer, "Detecting illegal logging with very high resolution satellites," *MAAP*, Report 125, Oct. 3, 2020. Available: `https://www.maapprogram.org/high-res-satellites/`. Accessed: Apr. 10, 2025.

[5] L. Villa and M. Finer, "Detecting logging in the Peruvian Amazon with high resolution imagery," *MAAP*, Report 94, Oct. 29, 2018. Available: `https://www.amazonconservation.org/detecting-logging-in-the-peruvian-amazon-with-high-resolution-imagery/`. Accessed: Apr. 10, 2025.

[6] S. Novoa, L. Villa, and M. Finer, "Detecting illegal logging with very high resolution satellites," *MAAP*, Report 125, Oct. 3, 2020. Available: `https://www.maapprogram.org/high-res-satellites/`. Accessed: Apr. 10, 2025.

Monitoring wildlife populations requires screening large conservation areas to locate animal groups or migration corridors, while high spatial resolution is needed to identify species and count individuals (Figure 9.4). Satellite observations can complement traditional monitoring approaches such as camera traps, acoustic sensing, UAV surveys, and field observations, which often suffer from limited spatial coverage or logistical constraints in remote or hazardous regions [42], [193].

A Tip and Cue architecture can support large-scale wildlife monitoring by combining wide-area search with targeted high-resolution follow-up imaging. High-resolution imagery can therefore support the detection of wildlife populations and improve the monitoring of population trends across inaccessible ecosystems.

Figure 9.4: Examples of satellite imagery showcasing WorldView-4 (0.31m/pix) wildlife imagery of African elephants. The right shows a zoomed in sample of an elephant, where the animal comprises around 15–20 pixels. [7]

Fish Spotting

Commercial fishing is profit-driven and depends on rapidly locating regions with the highest fish density. Fish schools are spatially dynamic and shift with ocean temperature, currents, and biological conditions, creating large-area search uncertainty.

Detecting fish schools therefore requires broad-area ocean monitoring to identify biological productivity indicators such as chlorophyll concentration or surface aggregation patterns. High spatial resolution is then needed to confirm school structure and concentration before vessels commit to a location.

A Tip and Cue architecture is relevant because it can reduce search time and uncertainty by combining wide-area scanning with targeted high-resolution observations. This can lower fuel consumption and operational costs while improving revenue and catch efficiency.

Rip Current Detection

Rip currents are narrow, fast-moving offshore flows that can form rapidly along coastlines and are highly localized and spatially unpredictable [196]. They are considered one of the most dangerous hazards for beachgoers worldwide. For example, in Australia rip currents cause more fatalities than floods, hurricanes, and tornadoes combined [197]. Their rapid formation and short lifetime make reliable forecasting difficult using conventional ocean and weather models [198], [199].

Detecting rip currents is challenging because they can appear suddenly, persist for short periods, and occur across large stretches of coastline. Their narrow spatial structure requires high spatial resolution to identify characteristic flow channels, while large coastal regions must be monitored to detect narrow surface flow anomalies before issuing warnings or closing affected zones (Figure 9.5).

Remote sensing methods using visible imagery, infrared observations, or SAR can reveal disruptions in wave-breaking patterns or variations in water surface roughness associated with rip currents [196], [200]. A Tip and Cue architecture could enable continuous monitoring of coastal regions to detect anomalous flow signatures, followed by targeted high-resolution observations to confirm rip channel geometry and support beach safety warnings.

[7] I. Duporge, O. Isupova, and S. Reece, "A new wildlife surveying technique: Using satellite imagery and machine learning to detect and monitor elephants," *Maxar Blog*, Dec. 3, 2019. Available: `https://blog.maxar.com/earth-intelligence/2019/a-new-wildlife-surveying-technique-using-satellite-imagery-and-machine-learning-to-detect-and-monitor-elephants`. Accessed: Apr. 10, 2025.

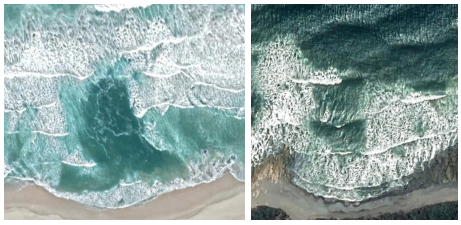


Figure 9.5: Examples of rip current detection from aerial and satellite imagery. [200]

Permanently Clouded Region Mapping

Persistent cloud coverage severely limits optical Earth observation in tropical and polar regions, preventing reliable monitoring of large areas of the Earth's surface [201], [202]. Some locations experience extremely high cloud occurrence, such as El Danubio (98.6%), Svalbard (96.7%), and Chuuk Lagoon (92%) [202]. As a result, optical sensors operating in the visible and infrared spectrum often fail to obtain usable observations. For example, Landsat imagery cannot monitor tropical rainforests in Brazil and Indonesia for more than ten months per year on average due to persistent cloud coverage [203]. This challenge is further compounded because cloud correction methods often rely on historical observations that may not be available [204].

Monitoring these regions therefore requires wide-area observation of cloud fields to identify short-lived gaps above areas of interest. The observational challenge is not only the persistence of clouds, but also the short duration of usable openings once they occur.

A Tip and Cue architecture can address this by using a wide-swath Tip sensor to continuously monitor cloud dynamics and detect transient openings, after which the Cue satellite can rapidly acquire high-resolution imagery before the gap closes. This approach is particularly valuable because observation opportunities in persistently clouded regions are often brief, increasing the importance of rapid follow-up imaging.

9.1.3. Urban and Industry

Methane Leak Detection

Methane ultra-emitters occur sporadically at industrial facilities involving oil and gas operations, pipelines, coal mines, and landfills [26], [205]. Although relatively rare, these events account for a significant share of global methane emissions, with equipment failures responsible for roughly 12% of total methane releases [26]. Rapid detection is therefore important for climate mitigation, as methane has a high short-term global warming potential.

Detecting these emissions requires wide-area spectral screening to identify anomalous methane plumes, while high spatial resolution is needed to attribute emissions to specific infrastructure (Figure 9.6). Current satellite systems either provide coarse global coverage, like Sentinel-5P with a spatial resolution of approximately 5.5×7 km, or high-resolution monitoring over small areas, such as GHGSat with observation windows of about 12×12 km [7], [206]. This creates a monitoring gap for detecting small methane leaks globally.

A two-scale observation architecture can address this gap by combining broad-area screening with targeted high-resolution follow-up imaging. This can improve both global coverage and precise source attribution.

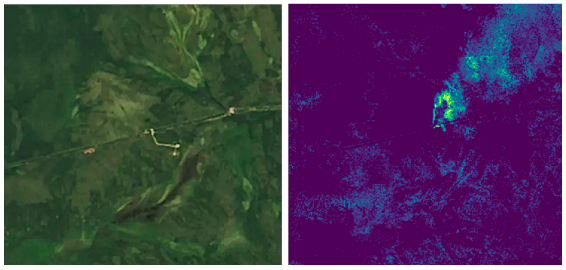


Figure 9.6: Examples of satellite imagery that shows a Russian gas pipeline (left) and a methane leak (right), where the methane leak was detected through spectral analysis. [8]

Oil Spill Detection

Oil spills pose a major environmental threat, impacting marine ecosystems, coastal regions, and local economies [207], [208]. Large spills are most commonly caused by tanker accidents transporting substantial quantities of oil. Globally, tanker incidents are estimated to release approximately 390 000 tonnes of oil into the ocean each year [209]. Rapid detection and localization are therefore critical for minimizing ecological damage and coordinating cleanup operations.

Detecting oil spills requires wide-area monitoring to identify anomalous surface reflectance or backscatter patterns associated with oil films, while high spatial resolution is needed to accurately map spill extent and identify the responsible vessel or infrastructure (Figure 9.7). Optical sensors, thermal infrared imagery, and Synthetic Aperture Radar (SAR) are commonly used for oil spill detection, as oil alters the surface roughness and reflective properties of the water [208].

A Tip and Cue architecture can support timely spill assessment by combining large-area monitoring with targeted high-resolution follow-up imaging. High-resolution satellite imagery can then guide response teams in deploying containment and recovery measures.

Figure 9.7: Examples of satellite imagery showing oil spills, floating on top of the oceanic surface. [9] [10] [11]

9.1.4. Natural Disasters

Wildfire Detection

Wildfires ignite unpredictably and can spread rapidly, causing severe environmental damage, economic losses, and human casualties [210]. In recent years, wildfire events have increased in scale and impact. For example, the 2025 California wildfires forced nearly 200,000 people to evacuate, caused approximately $20 billion in economic damage, and resulted in 27 fatalities [211]. Globally, almost four million square kilometers burned during the 2023–2024 fire season, with over 232,000 people evacuated in Canada alone [210]. Early detection is therefore critical for effective containment and evacuation planning.

[8] C. Katz, "Satellites seek out methane leaks from pipelines, oil fields, landfills and farms," *PBS NewsHour*, Jun. 17, 2021. Available: `https://www.pbs.org/newshour/science/satellites-seek-out-methane-leaks-from-pipelines-oil-fields-landfills-and-farms`. Accessed: Apr. 10, 2025.

[9] D. Werner, "NASA satellite data used to track Gulf of Mexico oil spill," *SpaceNews*, May 17, 2010. Available: `https://spacenews.com/nasa-satellite-data-used-track-gulf-mexico-oil-spill/`. Accessed: Apr. 10, 2025.

[10] La Redazione, "From satellites to the sea: predicting damage from oil spills," *Impactscool Magazine*, Apr. 7, 2020. Available: `https://magazine.impactscool.com/en/cambiamento-climatico-e-ambiente/dai-satelliti-al-mare-prevedere-i-danni-degli-sversamenti-di-petrolio/`. Accessed: Apr. 10, 2025.

[11] M. Carlowicz, "Satellites view California oil spill," *NASA Earth Observatory*, Oct. 3, 2021. Available: `https://earthobservatory.nasa.gov/images/148929/satellites-view-california-oil-spill`. Accessed: Apr. 10, 2025.

Detecting active fires requires wide-area monitoring to identify thermal anomalies, smoke plumes, or other indicators of combustion, while high spatial resolution is needed to map fire fronts and assess spread dynamics (Figure 9.8). Satellite instruments detect fires by measuring spectral radiance differences, particularly in infrared wavelengths where active fires strongly contrast with surrounding land surfaces [212].

A Tip and Cue architecture can support rapid wildfire response by combining broad-area hotspot detection with targeted high-resolution follow-up imaging. Rapid confirmation of detected hotspots improves response coordination and supports more effective deployment of firefighting resources.

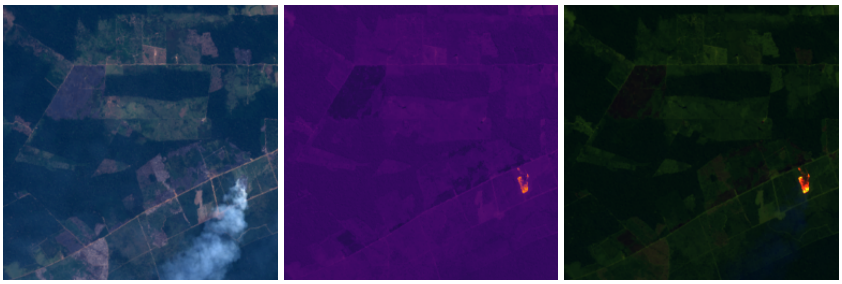


Figure 9.8: Examples of wildfire satellite imagery in different spectral band combinations, representing RGB (Band 2, 3, 4), IR (Band 5, 6, 7), and visible–IR combinations. [213]

Post-Disaster Damage Assessment

Immediately after disasters like earthquakes, floods, hurricanes, or wildfires, the spatial extent and severity of damage are often uncertain [30], [214], [215]. Rapid situational awareness is required for prioritizing rescue operations, assessing infrastructure damage, and planning evacuation routes.

Damage assessment requires wide-area satellite observations to identify the most affected regions, while high spatial resolution imagery is needed for detailed assessment of damaged buildings, flooded areas, and disrupted infrastructure (Figure 9.9). Pre-event satellite imagery further supports change detection, enabling direct comparison between pre- and post-disaster conditions to quantify damage.

A Tip and Cue architecture can support emergency response by combining broad-area identification of affected regions with targeted high-resolution follow-up imaging. This can improve rapid damage assessment and help guide rescue and recovery operations.

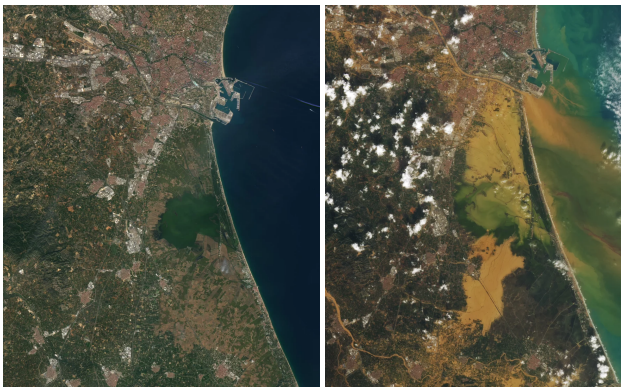


Figure 9.9: Example of pre- and post-event analysis for damage assessment, highlighting Landsat-8 images of the Valencia floods captured on Oct. 25, 2022 (left) and Oct. 30, 2024 (right). [12]

[12] J. Kim and M. Macias, "Satellite images show the devastation from Spain's deadly floods," *NPR*, Nov. 2, 2024. Available: `https://www.npr.org/2024/11/02/nx-s1-5176508/spain-valencia-chiva-flood-map`. Accessed: Apr. 14, 2025.

9.1.5. Space

Space Debris Monitoring

The number of tracked and untracked debris objects in Earth orbit continues to increase, raising collision risks for operational spacecraft [216]. Current estimates suggest that more than 30,000 debris objects larger than 10 cm and approximately 900,000 objects larger than 1 cm are orbiting the Earth [216]. According to the Kessler Syndrome, collisions between debris objects could generate further fragments and trigger a cascading chain reaction, significantly increasing the density of debris in orbit [217], [218].

Monitoring this environment requires wide-area observation across large orbital volumes to detect debris objects, while high spatial resolution is needed to characterize object size, shape, and attitude state (Figure 9.10). Most debris tracking is currently performed using ground-based radar and optical telescopes, which have limited fields of view and difficulty detecting small debris fragments [219].

Space-based observations can complement these systems by enabling wider coverage and improved object characterization. A Tip and Cue architecture could therefore support collision avoidance and conjunction assessment for operational satellites by combining broad-area detection with targeted high-resolution follow-up observations.

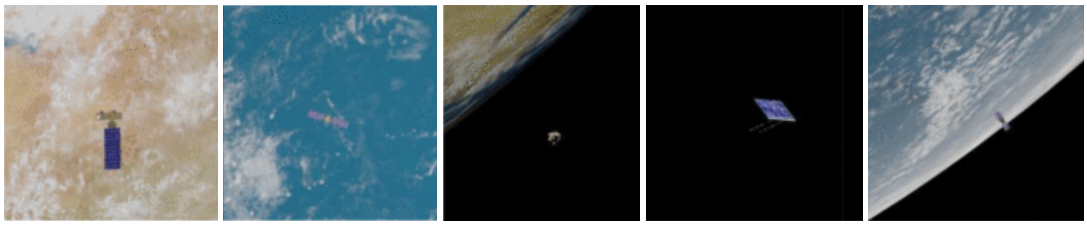


Figure 9.10: Examples of space debris imagery and simulated datasets used for detection and classification. [220]

9.2. Moderate Benefit

The applications in this section can benefit from AI-Based Tip and Cue under specific mission conditions, but the architectural advantage is less universal. In these cases, the value of targeted follow-up depends more strongly on latency requirements, sensing modality, and whether conventional scheduling can already address the observation need.

9.2.1. Security and Surveillance

Unauthorized Flight Detection

Unauthorized aircraft activity in restricted airspace poses risks to aviation safety and national security, particularly near sensitive infrastructure or controlled zones. Detecting aircraft that deviate from approved flight paths or operate without proper identification is therefore an increasingly important task for airspace authorities [221].

Monitoring these activities would require wide-area observations to detect anomalous flight behavior, while high spatial resolution could support aircraft identification.

However, radar networks and aircraft transponder systems already provide continuous tracking and identification of most aircraft. Because these ground-based systems deliver faster and more reliable monitoring, the structural advantage of a space-based Tip and Cue architecture is limited for unauthorized flight detection.

Illegal Land Activity Detection: Mining, Waste Disposal, Property Changes

Illegal land-use change from unauthorized mining, waste dumping, or property development often occurs in remote regions where ground inspections are difficult [222]. Illegal mining alone has significant environmental and social impacts, including pollution, tax evasion, and unsafe labor practices, with estimates suggesting that around one million children worldwide are involved in mining activities [222]. Satellite observations can therefore provide valuable insights by detecting changes in land cover or surface disturbances over large regions.

Monitoring these activities requires wide-area observations to reveal anomalies including excavation patterns, vegetation loss, or unusual surface materials, while high spatial resolution is needed to confirm illegal activities like waste dumping or unauthorized construction (Figure 9.11). For example, illegal waste disposal sites can be identified through characteristic surface patterns in satellite imagery [223], [224].

However, these activities typically develop over long time scales and rarely require rapid response. As a result, the operational advantage of a Tip and Cue architecture is reduced.

Figure 9.11: Examples of illegal waste dumpings from satellite imagery, where the red circles indicate suspicious objects. [13]

9.2.2. Ecosystem and Environment

Harmful Algal Bloom Detection

Harmful algal blooms (HABs) threaten marine ecosystems, public health, and coastal economies [225], [226]. In the United States alone, HABs are estimated to cause more than $81 million in economic losses annually and affect over 50% of U.S. states, mainly through impacts on aquaculture and fisheries [226]. Rising ocean temperatures and eutrophication are increasing the frequency and severity of these events worldwide [225].

Monitoring HABs requires observation of large coastal regions to detect abnormal chlorophyll concentrations and surface reflectance patterns, while higher spatial resolution imagery is needed to refine bloom boundaries and assess proximity to infrastructure (Figure 9.12). Satellite observations commonly detect HABs through spectral measurements of chlorophyll-a, which reflects green wavelengths while absorbing blue and red light [227], [228].

However, bloom development typically evolves over several days, reducing the urgency for rapid multi-scale observation compared to more time-critical applications. As a result, the structural advantage of a Tip and Cue architecture is moderate rather than high.

Figure 9.12: Examples of satellite imagery showcasing harmful algal blooms in coastal waters. [14] [15]

[13] R. N. Torres and P. Fraternali, "AerialWaste dataset for landfill discovery in aerial and satellite images," *Scientific Data*, vol. 10, no. 63, 2023. doi: `https://doi.org/10.1038/s41597-023-01976-9`.

[14] National Oceanic and Atmospheric Administration, "What is harmful algal bloom?," *NOAA*. Available: `https://www.noaa.gov/what-is-harmful-algal-bloom`. Accessed: Apr. 10, 2025.

[15] A. H. Schmitt, "Downstream consequences: How NASA satellites track harmful algal blooms," *NASA*, Jan. 21, 2021. Available: `https://www.nasa.gov/missions/aqua/downstream-consequences-how-nasa-satellites-track-harmful-algal-blooms/`. Accessed: Apr. 10, 2025.

Sargassum Monitoring
Sargassum blooms consist of large floating mats of brown algae that can accumulate along coastlines, affecting tourism, fisheries, and coastal infrastructure [229]. When decomposing on beaches, these blooms release hydrogen sulfide and ammonia, which can pose health risks and environmental hazards.

Monitoring sargassum requires large-scale ocean observation to detect and track these drifting accumulations across ocean basins, while higher spatial resolution imagery can refine shoreline impact assessments.

However, the movement of sargassum is primarily governed by ocean circulation and seasonal patterns, and response planning typically occurs over multi-day to weekly time scales. As a result, the lower latency sensitivity reduces the structural advantage of a Tip and Cue architecture compared to more time-critical applications.

Biodiversity Monitoring
Biodiversity monitoring supports long-term ecosystem stability assessment and climate policy reporting [230]. Globally, around 73,000 tree species have been identified, with estimates suggesting that approximately 9,000 species remain undiscovered [230]. Understanding species diversity is required for assessing ecosystem resilience, as declining biodiversity can increase vulnerability to ecosystem collapse and reduce critical services inlcluding clean water provision, climate regulation, and pollination.

Monitoring biodiversity with satellite observations relies on detecting spectral variations in vegetation canopies that correspond to differences in plant species composition (Figure 9.13). High-resolution imagery can improve vegetation structure analysis and assist in identifying biodiversity patterns across large forested regions.

However, biodiversity monitoring typically relies on long-term trend analysis rather than rapid detection of discrete events, and low-resolution imagery often lacks clear features that would trigger targeted follow-up observations. As a result, the structural advantage of a Tip and Cue architecture is limited for this application.

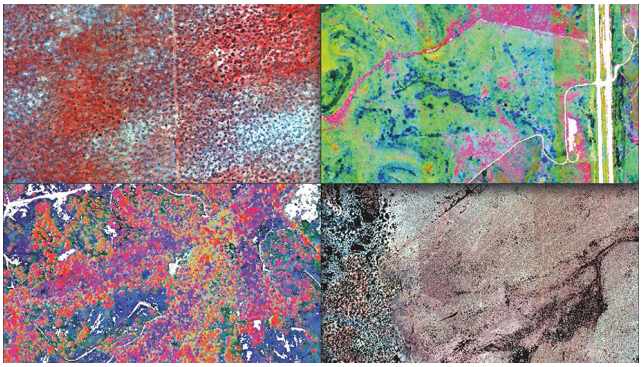


Figure 9.13: Satellite imagery reflecting biodiversity, where different plant species are represented as spectral variations. [16]

Plastic Soup Mapping
Marine litter poses a significant environmental threat, affecting coastal ecosystems, fisheries, and human health [231]. Current estimates indicate that between 75 and 199 million tonnes of litter are present in the oceans, with plastics accounting for roughly 85% of the total and an additional 13 to 37 million tonnes expected to enter the oceans annually by 2040 [231]. Ocean currents transport floating debris into elongated accumulation patterns that can span large oceanic regions.

Monitoring marine litter requires wide-area observation to detect these accumulation zones, while higher spatial resolution imagery improves mapping accuracy and supports the identification of dense debris patches (Figure 9.14). Satellite observations have demonstrated that marine litter can be detected through characteristic spatial patterns in high-resolution imagery.

[16] A. K. Schweiger and E. Laliberté, "Satellite monitoring of biodiversity moves within reach," *University of Zurich*, May 19, 2022. Available: `https://www.news.uzh.ch/en/articles/media/2022/Biodiversity-Monitoring.html`. Accessed: Apr. 10, 2025.

However, plastic drift evolves over weeks to months, and mitigation measures are rarely time-critical at the scale of hours or days. As a result, the operational advantage of a rapid Tip and Cue architecture is limited for this application.

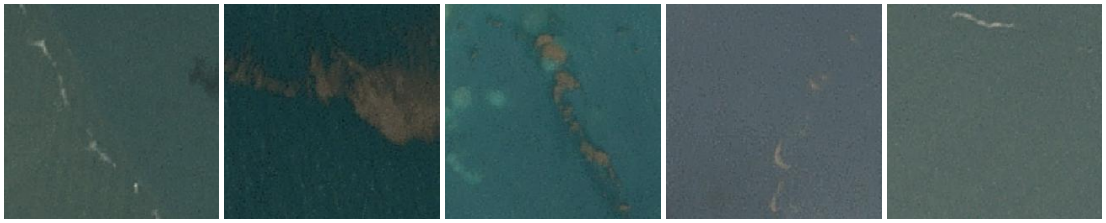


Figure 9.14: Examples of marine litter captured by PlanetScope satellite imagery with 3 m spatial resolution, where floating debris forms elongated patterns shaped by ocean currents. [17]

9.2.3. Urban and Industry

Emission Profile Monitoring

Regional emission monitoring supports climate reporting, regulatory compliance, and industrial planning [232], [233]. Accurate measurements of pollutants CO_2, NH_3, and NO_x are essential for assessing whether industries and agricultural activities comply with environmental regulations and sustainability targets [232]. Insufficient spatial resolution in current monitoring networks can lead to large uncertainties in emission attribution and policy enforcement.

Monitoring these emissions requires wide-area observation to detect regional patterns, while higher spatial resolution is needed to attribute emissions to individual facilities or industrial zones.

However, emission sources are typically continuous and evolve over longer time scales rather than appearing as short-lived events. In addition, atmospheric gas monitoring relies strongly on spectral measurements that may degrade at off-nadir viewing angles, reducing the structural advantage of a Tip and Cue architecture compared to nadir-only atmospheric observation systems.

Sewer Inspection

Urban sewer networks are critical infrastructure for sanitation, flood prevention, and public health. However, many sewer systems were built after World War II with an expected lifespan of roughly 80 years. As a result, large portions of underground infrastructure in cities worldwide are approaching the end of their technical lifetime [234]. Aging sewer systems are therefore increasingly vulnerable to structural failures, blockages, or leakage, which can lead to flooding, environmental contamination, and costly infrastructure damage, particularly in densely populated urban areas [235].

Satellite observations could support sewer infrastructure monitoring by detecting indirect indicators of underground failures, displaying localized ground deformation, surface moisture anomalies, or subsidence patterns above sewer lines. High-resolution imagery may help identify areas where infrastructure degradation affects the surface environment.

However, sewer systems are underground and failures are typically detected more directly through ground-based inspection tools, sensors, or maintenance reports. Because satellite observations provide only indirect indicators and inspections can be triggered from ground monitoring systems, the structural advantage of a Tip and Cue architecture is limited for sewer inspection.

Agriculture Monitoring

Agricultural productivity is increasingly threatened by climate change, land degradation, and crop diseases [236]. Global crop yields are projected to decline by 11–25% by the end of the century, increasing the risk of food insecurity [236]. Soil salinisation alone affects approximately one billion hectares of land worldwide, particularly in arid and semi-arid regions [237]. Crop diseases can also cause major losses, for example, black pod disease accounts for roughly 44% of global cacao crop losses and can spread through plant tissue within several days [238], [239].

[17] A. Shah, L. Thomas, and M. Maskey, "Marine Debris Dataset for Object Detection in Planetscope Imagery," Version 1.0, *Radiant MLHub*, 2021. doi: `https://doi.org/10.34911/rdnt.9r6ekg`.

Satellite observations support agricultural monitoring by detecting changes in vegetation reflectance, chlorophyll content, and canopy structure using visible and near-infrared spectral bands [240]. These measurements can reveal crop stress, disease spread, or soil degradation patterns across large agricultural regions (Figure 9.15).

However, crop monitoring typically occurs over seasonal time scales, and affected areas are often already known to farmers or local authorities. As a result, the urgency-driven advantage of a rapid Tip and Cue architecture is limited compared to applications requiring immediate detection and response.

Figure 9.15: Examples of false color satellite imagery for agricultural monitoring, where different band combinations reveal variations in vegetation health and crop conditions. [18]

9.2.4. Natural Disasters

Flooding Dynamics Mapping

Flooding risks are increasing globally due to sea-level rise and extreme precipitation events [241]. Low-lying regions are particularly vulnerable, for example, approximately 55% of the Netherlands is susceptible to flooding, requiring careful evacuation planning and infrastructure protection [241]. Understanding how floodwaters propagate through urban and coastal environments is therefore important for risk mitigation and emergency response planning.

Satellite observations can provide large-scale monitoring of flood extent and surface water dynamics, complementing hydrodynamic simulation models that estimate water flow and inundation patterns [242]. High spatial resolution imagery can improve the mapping of flood boundaries and water movement around infrastructure.

However, flood-prone regions are usually known in advance and monitoring is often triggered by predicted weather events or infrastructure failures. As a result, the structural advantage of a Tip and Cue architecture is limited, since targeted observations can typically be scheduled without requiring continuous wide-area tipping.

Avalanche Prediction

Avalanches pose a major hazard in mountainous regions, threatening infrastructure, transportation routes, and human safety [243]. Their occurrence is influenced by terrain steepness, snow accumulation, and temperature fluctuations, with climate change expected to increase variability in snowfall and snowpack stability. Monitoring avalanche-prone regions is therefore important for risk management in mountain communities and recreational areas.

Satellite observations can support avalanche detection by analyzing terrain characteristics and monitoring rapid changes in snow cover using optical or radar imagery (Figure 9.16). Synthetic Aperture Radar (SAR) in particular enables detection of avalanche scars and snow displacement patterns even under cloudy conditions [243].

However, avalanche-prone locations are typically well known and monitoring efforts are often triggered by local weather forecasts or ground-based measurements. As a result, the operational advantage of a Tip and Cue architecture is limited compared to targeted observations in predefined high-risk areas.

[18] U.S. Geological Survey, "What the colors mean," *Earthshots, EROS Center*. Available: `https://eros.usgs.gov/earthshots/what-the-colors-mean`. Accessed: Apr. 14, 2025.

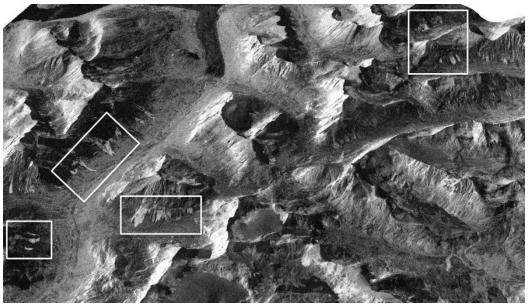


Figure 9.16: Example of avalanches detected with SAR in Troms County in northern Norway, where the white squares indicate avalanche areas. [19]

Hurricane Prediction

Hurricanes are among the most destructive natural disasters, causing large human and economic losses [31]. For example, Hurricane Helene in September 2024 resulted in more than 230 fatalities across six southeastern U.S. states and caused nearly $60 billion in damage while destroying over 73,000 homes [244], [245]. Early detection and accurate prediction are therefore critical for timely evacuation planning and disaster preparedness.

Satellite observations play a central role in hurricane monitoring by detecting cloud formations, convection patterns, and ocean temperature anomalies that indicate developing tropical cyclones (Figure 9.17). Visible and infrared imagery reveal storm structure and cloud-top temperatures, while microwave and SAR observations provide insight into precipitation patterns and wind fields [246].

However, hurricane prediction relies heavily on numerical weather models that integrate multiple atmospheric data sources and large-scale observations. Since regions of interest are typically identified through these models, targeted satellite observations can already be scheduled in advance, limiting the operational advantage of a Tip and Cue architecture.

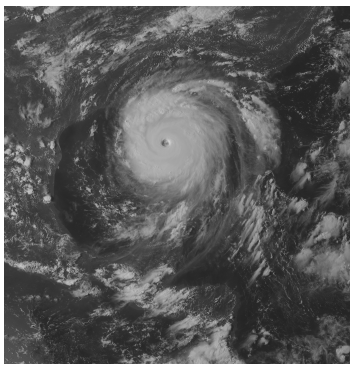


Figure 9.17: Example of a raw satellite image from GOES (visible + IR), showing Hurricane Katrina approaching the Louisiana coast. [20]

9.2.5. Space

Solar Flare and Coronal Mass Ejection Monitoring

High-energy solar flares and coronal mass ejections (CMEs) can disrupt satellite communications, navigation systems, and terrestrial power grids, potentially causing severe economic damage and infrastructure failures [247]. Reliable forecasting is therefore needed to allow preventive actions like temporarily shutting down vulnerable satellite systems or adjusting spacecraft operations.

[19] N. Bazilchuk, "Detecting avalanches with satellite radar," *Science Norway*, Dec. 14, 2015. Available: `https://www.sciencenorway.no/climate-forskningno-geology/detecting-avalanches-with-satellite-radar/1426782`. Accessed: Apr. 14, 2025.

[20] NASA Goddard Space Flight Center, "Hurricane Katrina, August 28, 2005," Flickr, Aug. 29, 2008. Available: `https://www.flickr.com/photos/gsfc/20439378136`. Accessed: Apr. 14, 2025.

Space-based observatories monitor the Sun to detect magnetic activity, energetic particles, and plasma eruptions associated with solar events (Figure 9.18). Instruments such as the Extreme Ultraviolet Imager aboard the Solar Orbiter capture high-resolution observations of solar structures and eruption processes [248].

However, solar monitoring missions typically observe the entire solar disk continuously from dedicated heliocentric platforms. Because the region of interest is always known and observation latency requirements are relatively moderate, combining wide-field and high-resolution measurements on a single spacecraft is generally more efficient than deploying a separate Tip and Cue architecture.

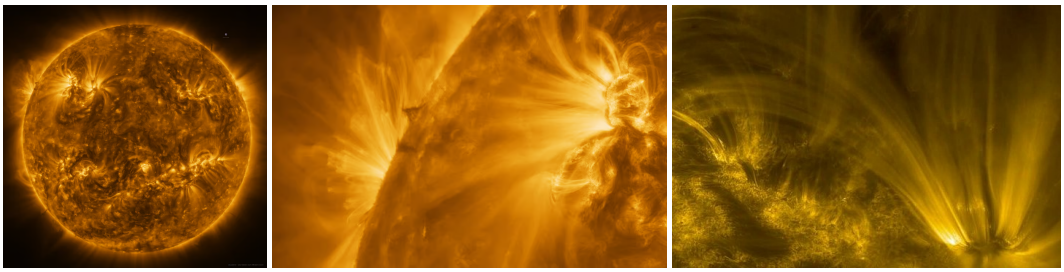


Figure 9.18: Images captured by the Solar Orbiter showing coronal mass ejections and solar surface activity. [21] [22]

Satellite Inspection

The growing number of operational spacecraft and space debris has increased the need for reliable satellite inspection capabilities [249]. Structural damage caused by debris impacts, launch anomalies, or component degradation can reduce mission performance or lead to premature satellite failure. On-orbit servicing concepts aim to address these challenges by enabling inspection, repair, refueling, and upgrading of spacecraft already operating in orbit [217], [250].

High-resolution space-based observations could support the identification of damaged components like solar panels or structural elements. Close-proximity inspection missions have therefore been proposed, including CubeSat swarms designed to monitor larger spacecraft or space stations [251].

However, inspection targets are typically known in advance and require controlled proximity operations rather than wide-area search. As a result, the architectural advantage of a Tip and Cue configuration is limited compared to dedicated inspection spacecraft designed specifically for close-range observation.

Landing Site Assessment

Safe landing site selection is a critical component of future exploration missions to the Moon and Mars [252]. Autonomous landing systems such as NASA's Autonomous Landing and Hazard Avoidance Technology (ALHAT) and Safe and Precise Landing (SPLICE) programs use onboard image processing to identify safe terrain during descent [253]. However, orbital terrain maps often lack the spatial resolution required to detect small hazards such as rocks, craters, or steep slopes within the limited time available for landing decisions [254].

High-resolution orbital imagery can improve the mapping of potential landing zones and surface hazards (Figure 9.19). Missions such as Tianwen-1 have already demonstrated the capability to capture sub-meter resolution images of the Martian surface from orbit [255].

Nevertheless, landing operations are typically supported by sensors directly integrated into the lander itself, enabling immediate hazard detection and navigation during descent. Because the region of interest is predetermined and close-range imaging is required, combining wide-field and high-resolution sensors on the lander is generally more efficient than deploying a separate Tip and Cue satellite architecture.

[21] N. Lavars, "Solar Orbiter probe snaps the sharpest ever image of the Sun's corona," *New Atlas*, Mar. 24, 2022. Available: `https://newatlas.com/space/solar-orbiter-probe-sharpest-image-sun-corona/`. Accessed: Apr. 14, 2025.

[22] European Space Agency, "Solar Orbiter's first close encounter: Latest Solar Orbiter images and videos, May 2022," May 2022. Available: `https://www.esa.int/esatv/Videos/2022/05/Solar_Orbiter_s_first_close_encounter/Latest_Solar_Orbiter_images_and_videos_May_2022`. Accessed: Apr. 14, 2025.

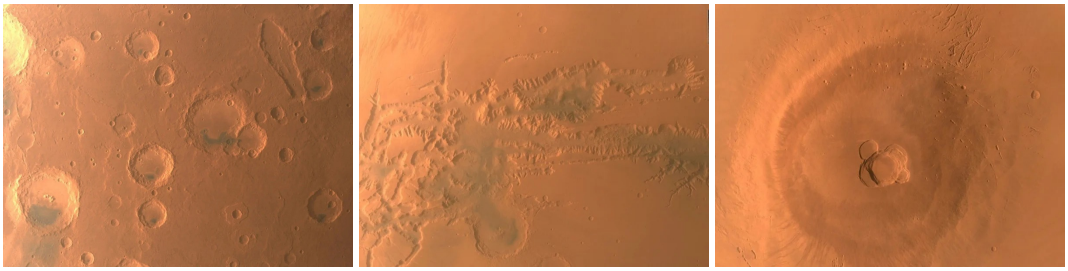


Figure 9.19: Images captured by the Tianwen-1 mission showing high-resolution observations of the Martian surface. [23] [24]

9.3. Limited Benefit

The applications in this section show limited expected benefit from AI-Based Tip and Cue. Either the region of interest is already known in advance, the latency requirement is less strict, or the sensing task is more efficiently handled by conventional wide-swath or single-platform observations.

9.3.1. Security and Surveillance

Accident Investigation

Satellite imagery can support accident investigation by providing independent observations of remote or inaccessible locations [256]. High-resolution imagery has been used to analyze transportation incidents and large-scale disasters, including the investigation of the MH17 aircraft crash in 2014, where satellite observations were used to identify possible launch locations and trajectories (Figure 9.20) [257]. Such observations can complement ground-based reports and help reconstruct accident circumstances.

Accident investigation requires high-resolution imagery to map accident sites and surrounding infrastructure in detail. However, accident locations are typically identified rapidly through eyewitness reports, emergency calls, or radar tracking systems.

As a result, targeted satellite observations can usually be scheduled directly without requiring wide-area search. The limited need for broad scanning therefore reduces the architectural advantage of a Tip and Cue configuration for accident investigation.

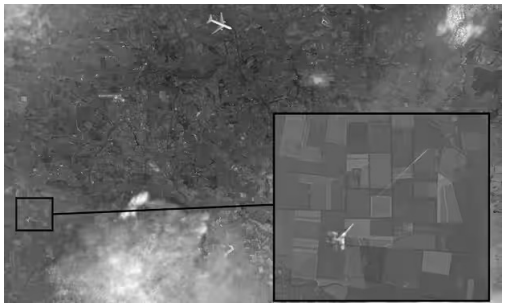


Figure 9.20: Example of satellite imagery used during the investigation of the MH17 aircraft incident. [25]

[23] T. Zhang, "It's a wrap for China's Tianwen-1 Mars mission, but rover and orbiter still ready to work," *South China Morning Post*, Jun. 29, 2022. Available: `https://www.scmp.com/news/china/science/article/3183557/its-wrap-chinas-tianwen-1-mars-mission-rover-and-orbiter-still`. Accessed: Apr. 14, 2025.

[24] M. Growcoot, "Detailed photos of Mars captured by China's Tianwen-1 probe," *PetaPixel*, Jun. 29, 2022. Available: `https://petapixel.com/2022/06/29/detailed-photos-of-mars-captured-by-chinas-tianwen-1-probe/`. Accessed: Apr. 14, 2025.

[25] The Guardian, "US dismisses Russian MH17 pictures that blame Ukraine for disaster", Nov. 15, 2014. Available: `https://www.theguardian.com/world/2014/nov/15/ukraine-fighter-shot-mh17-claims-russian-tv-photo-fake`. Accessed: Apr. 10, 2025.

9.3.2. Ecosystem and Environment

Coral Reef Monitoring

Coral reefs are among the most biodiverse marine ecosystems and are increasingly threatened by climate change and rising sea temperatures. Coral bleaching events, caused by prolonged thermal stress, can lead to large-scale reef degradation and loss of marine biodiversity. Monitoring reef health is therefore important for understanding ecosystem changes and supporting conservation efforts.

Satellite observations can detect changes in sea surface temperature and reef reflectance patterns, enabling large-scale monitoring of coral reef conditions (Figure 9.21). High-resolution imagery can further reveal spatial patterns of reef degradation and recovery.

However, coral bleaching and reef degradation typically occur over longer time scales rather than requiring rapid detection and response. As a result, the urgency-driven advantage of a Tip and Cue architecture is limited for this application.

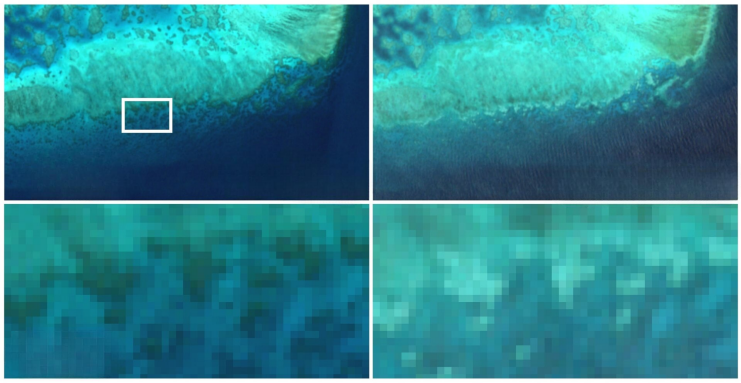


Figure 9.21: Satellite imagery of the Great Barrier Reef captured on 8 June 2016 and 23 February 2017, highlighting reef degradation over time. [26]

Ocean Dead Zone Monitoring

Ocean dead zones are regions with extremely low oxygen concentrations that threaten marine ecosystems and fisheries. These hypoxic zones can develop due to nutrient pollution, ocean warming, and biological activity that reduces dissolved oxygen levels in coastal waters [258].

Satellite observations can support the monitoring of environmental indicators associated with dead zone formation, like temperature and sulfide concentrations across large ocean regions [259].

However, dead zone development typically occurs gradually over seasonal time scales rather than as rapid events requiring immediate response. Consequently, the operational advantage of a rapid Tip and Cue architecture is limited compared to long-term environmental monitoring approaches.

Glacier Dynamics Monitoring

Glacier and ice sheet dynamics are important indicators of climate change and major contributors to global sea-level rise. The Antarctic Ice Sheet alone contains approximately 91% of the Earth's total ice mass, making its evolution critical for long-term sea-level projections [260]. Monitoring glacier retreat, ice velocity, and surface melting therefore provides important information for climate research and environmental policy.

Satellite observations enable large-scale monitoring of glacier extent and surface changes over long time periods (Figure 9.22). High-resolution imagery can reveal variations in glacier fronts, ice flow patterns, and the formation of glacial lakes.

[26] European Space Agency, "Sentinel-2 captures coral bleaching of Great Barrier Reef," May 24, 2017. Available: `https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-2/Sentinel-2_captures_coral_bleaching_of_Great_Barrier_Reef`. Accessed: Apr. 10, 2025.

However, glacier dynamics evolve gradually over years to decades, and monitoring typically focuses on repeated observations of known locations rather than rapid detection of new events. Consequently, the structural advantage of a Tip and Cue architecture is not utilized for this type of long-term environmental monitoring.

Figure 9.22: Satellite imagery of the Coast Range glacier captured on September 26, 1984 (left) and September 22, 2023 (right), illustrating glacier retreat and the formation of a glacial lake. [27]

Iceberg Detection

Icebergs pose potential hazards for maritime navigation and are important indicators of polar ice dynamics [261]. While large icebergs are regularly monitored, smaller icebergs are more difficult to track despite their relevance for understanding ice mass balance and regional ocean processes [262]. Monitoring iceberg populations therefore contributes to both climate studies and maritime safety.

Satellite observations enable large-area monitoring of polar oceans and can detect drifting icebergs across remote regions. High-resolution imagery can further refine iceberg size, trajectory, and melting patterns.

However, iceberg movement typically occurs gradually and can also be monitored using ship-based radar and existing maritime surveillance systems. Because detection does not require immediate response and alternative monitoring infrastructure already exists, the structural advantage of a Tip and Cue architecture is limited for this application.

9.3.3. Urban and Industry

Power Grid Failure Detection

Reliable electricity infrastructure is crucial for economic activity, transportation, and communication systems [263]. In many low- and middle-income countries, power outages remain common and can significantly hinder economic development and social stability. Mapping the spatial distribution and duration of electricity disruptions is therefore important for infrastructure planning and disaster response [264].

Satellite observations can detect large-scale power outages by monitoring nighttime light intensity and identifying sudden reductions in illumination across urban areas. High-resolution imagery can then provide additional detail on affected regions and infrastructure.

However, power outages are typically identified through ground-based monitoring systems or reports from local authorities, after which targeted satellite observations can be scheduled. As a result, the structural advantage of a Tip and Cue architecture is limited for this application.

Ground Movement and Geohazards

Ground movement and geohazards like soil subsidence, landslides, and sinkholes can threaten infrastructure including dams, bridges, buildings, and transportation networks. Monitoring terrain stability is therefore important for preventing structural failures and managing risks in vulnerable regions.

Satellite observations can detect ground deformation and terrain changes through repeated measurements of surface displacement and vegetation or moisture variations. High-resolution imagery can help identify areas where infrastructure or natural terrain may become unstable.

[27] A. Voiland, "Retreat at Klinaklini," *NASA Earth Observatory*, Jan. 29, 2024. Available: `https://earthobservatory.nasa.gov/images/152378/retreat-at-klinaklini`. Accessed: Apr. 10, 2025.

However, these processes typically evolve gradually over long periods of time, and monitoring often focuses on predefined high-risk locations using ground sensors or targeted satellite observations. As a result, the operational advantage of a Tip and Cue architecture is limited for this type of geohazard monitoring.

Traffic Flow Optimization
Many metropolitan areas measure vehicle counts on highways and selected major roads, but monitoring coverage remains limited because permanent ground-based traffic stations are expensive to deploy [265]. Such data supports urban planning, congestion management, and supply chain logistics. Satellite imagery can complement these systems by providing large-scale observations of transportation networks, ports, and shipping lanes, enabling the identification of traffic density patterns across wide geographic regions.

High-resolution imagery can improve the mapping of vehicle or vessel distributions in congested areas. However, traffic hotspots like highways, urban centers, and major ports are generally known in advance, allowing targeted observations without the need for wide-area search.

Consequently, the structural advantage of a Tip and Cue architecture is limited for traffic flow monitoring.

Archaeological Site Detection
Tip and Cue satellites can aid in the exploration and protection of cultural heritage in remote deserts and underwater. To date, only 19% of the world's oceans have been mapped, leaving most of the marine environment, including shipwreck sites, undiscovered. Estimates indicate that around three million shipwrecks lie on the ocean floor globally [266]. Remote sensing data was used earlier to explore the Titanic wreck, contributing to forensic investigations and deepwater archaeological research of shipwreck sites [267]. On land, earth observation data from Tunisia enabled the reconstruction of a military defense system with Roman forts, which provides new insights into the strategy of the Roman empire [268]. Figure 9.23 provides an example of archaeological site detection from space.

Tip and Cue can help overcome challenges such as hard-to-access geography and a large spread of probable locations. A Tip satellite could detect broad surface-level variations that suggest archaeological sites, after which a Cue satellite could provide the high-resolution imaging needed to identify structures like shipwrecks, submerged cities, or buried remains.

However, archaeological site detection does not require a critical time response, and small remains may be hard to detect in low-resolution imagery. Therefore, the structural advantage of a Tip and Cue architecture remains limited.

Figure 9.23: Example of archaeological sites detected from space, showcasing Roman forts in the hard-to-access conflict zones Syria and Iraq. [28]

Remote Trash Mapping
Accumulated waste in remote natural environments is becoming an increasing environmental concern. For example, litter densities on Bali's beaches are estimated at approximately 7.15 pieces per square

[28] J. Casana, "Spy satellites reveal hundreds of undiscovered Roman forts," *Phys.org*, Oct. 27, 2023. Available: `https://phys.org/news/2023-10-spy-satellites-reveal-hundreds-undiscovered.html`. Accessed: Apr. 14, 2025.

meter, with plastics and foam representing the majority of debris [269]. Even high-altitude locations such as Mount Everest are affected, where discarded equipment from previous expeditions has accumulated over decades [270].

Satellite observations can support environmental monitoring by detecting large debris accumulations in remote or difficult-to-access regions. High-resolution imagery can help map waste concentrations and identify areas where environmental degradation occurs.

However, such waste accumulation typically develops gradually and known hotspots are often located near established camps or settlements. Because detection does not require rapid response and targeted observations can already be planned, the structural advantage of a Tip and Cue architecture is limited for remote trash mapping.

Solar Panel Inspection

Large solar panel installations are increasingly deployed on rooftops and in utility-scale solar farms, requiring regular inspection to ensure optimal power production. Damage, dirt accumulation, or malfunctioning cells can reduce energy output and lead to operational inefficiencies. Remote sensing techniques have been explored to detect these anomalies across large solar arrays [271].

Satellite observations can support the monitoring of solar installations by detecting structural patterns and surface changes across large areas. High-resolution imagery can help identify damaged or poorly performing panel sections (Figure 9.24).

However, degradation processes typically occur gradually, and power production data already provides direct indicators of malfunction. Because inspection can be scheduled once anomalies are detected from operational data, the structural advantage of a Tip and Cue architecture is limited for solar panel monitoring.

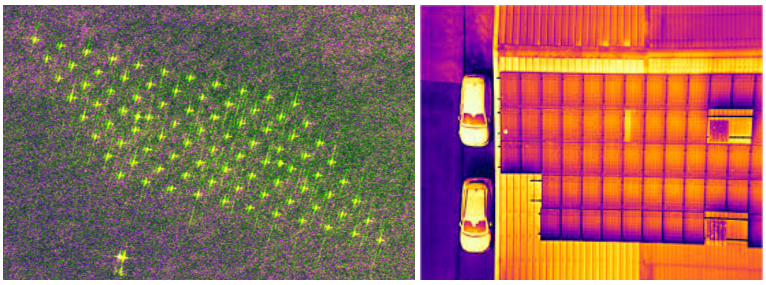


Figure 9.24: Inspection of windmill orientation through SAR imagery (left) and solar panel malfunction detection through aerial imagery (right), where the increased brightness highlights a damaged cell. [29] [30]

Wind Turbine Inspection

Offshore wind turbines operate in harsh marine environments characterized by high humidity, salinity, and strong winds, which can accelerate structural degradation and increase the risk of operational failures [271]. Regular inspection is therefore required to ensure safe operation and maintain optimal energy production.

Satellite imagery can support the monitoring of large offshore wind farms by detecting turbine orientation, structural changes, or surrounding environmental conditions. High-resolution observations can provide additional insight into turbine alignment and operational status.

However, wind farm locations are known in advance and turbine performance is continuously monitored through operational data. As a result, targeted inspections can already be scheduled when anomalies occur, limiting the structural advantage of a Tip and Cue architecture for wind turbine monitoring.

[29] N. Keseric and R. Hall, "An eye in the sky for wind turbines," *Loop, Equinor*, Oct. 18, 2018. Available: `https://loop.equinor.com/en/stories/satellite-images-turbines`. Accessed: Apr. 10, 2025.

[30] "Detecting issues early with thermal inspections of domestic solar panels," *Drone Media Imaging*. Available: `https://www.dronemediaimaging.co.uk/detecting-issues-early-with-thermal-inspections-of-domestic-solar-panels/`. Accessed: Apr. 10, 2025.

Thermal Insulation Monitoring
Building heating represents a significant contributor to greenhouse gas emissions. For example, a case study in Latvia found that buildings account for more than 45% of the country's final energy consumption, with over 70% of this demand originating from residential buildings [272]. Improving thermal insulation therefore plays an important role in reducing energy consumption and emissions.

Thermal infrared satellite imagery can support the identification of heat loss in urban areas by detecting abnormal surface temperature patterns across rooftops and building facades. High-resolution observations could help identify poorly insulated structures and guide energy-efficiency improvements.

However, insulation deficiencies typically persist over long time periods and do not require rapid detection or response. Because observations can be scheduled and repeated without strict timing constraints, the structural advantage of a Tip and Cue architecture is limited for thermal insulation monitoring.

Livestock Management
Monitoring livestock across large grazing areas can be challenging, particularly in remote or difficult-to-access regions. Efficient tracking of animal distribution is important for both agricultural management and environmental policy enforcement. In the Netherlands, for example, farmers receive subsidies to support meadow bird habitats and biodiversity, which requires careful management of grazing intensity. However, inspections conducted by the Dutch RVO currently cover only about 5% of these areas and typically occur once or twice per year [273]. Similar monitoring challenges occur in mountainous regions such as the Alps, where steep terrain, harsh weather, and large grazing areas complicate direct observation [274].

Satellite imagery can support livestock monitoring by providing periodic observations of grazing areas and animal distributions. High-resolution imagery may enable the identification of livestock presence and grazing patterns across large landscapes.

However, livestock monitoring generally does not require rapid detection or response, and observations can be scheduled based on management needs. Because targeted imaging can already be requested when needed, the structural advantage of a Tip and Cue architecture is limited for livestock monitoring.

9.3.4. Natural Disasters

Seismic Activity: Tsunamis, Earthquakes
Seismic hazards, earthquakes and tsunamis, are among the most destructive natural disasters, capable of causing large-scale infrastructure damage and loss of life [275]. Tsunamis occur less frequently than many other natural hazards but can devastate coastal regions within minutes after an earthquake, often with little warning [276]. Satellite observations can contribute to seismic monitoring by measuring ground deformation with Interferometric Synthetic Aperture Radar (InSAR) and detecting surface changes or atmospheric anomalies that may precede seismic events [277], [278]. These observations are typically combined with other monitoring systems like GNSS networks, gravity measurements, and ocean-based sensors including Deep-ocean Assessment and Reporting of Tsunamis (DART) buoys [279], [280].

Monitoring seismic hazards with satellite observations requires high radiometric sensitivity to detect subtle variations in ground deformation or related signals [281]. Earthquake occurrence is also strongly associated with known tectonic plate boundaries, allowing monitoring systems to focus observations on predefined high-risk regions [282].

Because these processes evolve over longer time scales and observations can be scheduled in advance, the structural advantage of a Tip and Cue architecture is limited for seismic activity monitoring.

Volcanic Eruption Prediction
Remote sensing plays an important role in volcanic eruption monitoring by observing activity indicators like ground deformation, gas emissions, thermal anomalies, and surface changes [283]. Techniques including Interferometric Synthetic Aperture Radar (InSAR) and GNSS measure ground deformation caused by magma movement beneath the surface, while thermal infrared sensors detect heat anomalies linked to increased volcanic activity [284], [285], [286]. Hyperspectral and ultraviolet observations can also monitor volcanic gas emissions including sulfur dioxide, which often increase prior to eruptions [287].

Monitoring volcanic activity with satellite observations focuses on known volcano locations and geothermal hotspots, allowing targeted measurements of deformation, gas release, and thermal change.

Because these regions of interest are known in advance, observations can be scheduled directly without the need for wide-area search, and many monitoring systems already combine satellite and ground-based measurements for early warning. The structural advantage of a Tip and Cue architecture is therefore limited for volcanic eruption monitoring.

Lightning Strike Mapping
Lightning monitoring is important for understanding severe weather formation, wildfire ignition, infrastructure damage, and aviation safety [288]. Global lightning activity is currently monitored by networks like the World Wide Lightning Location Network (WWLLN), which uses approximately 70 very low frequency radio sensors to detect lightning discharges worldwide [289]. However, coverage and localization accuracy remain limited over oceans, polar regions, and other remote locations.

Satellite observations could complement ground-based networks by detecting lightning activity over poorly monitored regions and providing improved spatial coverage.

However, lightning discharges occur on millisecond timescales and require extremely high temporal resolution to observe individual events. Because detection relies primarily on specialized sensors designed for rapid signal capture rather than spatial cueing, the structural advantage of a Tip and Cue architecture is limited for lightning strike monitoring.

Transient Luminous Event Mapping
Transient luminous events (TLEs), including sprites, elves, and blue jets, are short-lived optical phenomena occurring in the upper atmosphere above thunderstorms [290]. These events originate from unusual lightning discharges in the mesosphere and are of increasing interest for atmospheric and climate research. Current observations from missions such as the Global Lightning and Sprite Measurement Mission aboard the International Space Station provide valuable data, but the available spatial resolution of approximately 0.55 km per pixel limits detailed analysis [291].

Although TLEs occur at unpredictable locations, they evolve on millisecond timescales, making them extremely difficult to capture using conventional Earth observation systems.

Detecting these phenomena therefore requires dedicated high-speed sensing rather than multi-scale spatial cueing. As a result, the structural advantage of a Tip and Cue architecture is limited for transient luminous event monitoring.

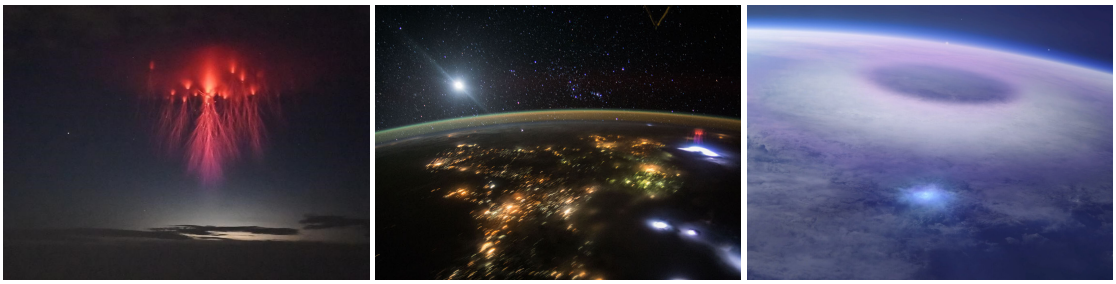


Figure 9.25: From left to right: Sprites captured from Earth; sprites and elves captured from space; elves captured from space. [31] [32] [33]

9.3.5. Space

Aurora Borealis Monitoring
Aurora observations provide important insight into space weather processes and the interaction between the solar wind, magnetosphere, and ionosphere during geomagnetic storms [292]. These observations

[31] S. Hummel, "Sprites, Elves, Ghosts & More: Photographing Transient Luminous Events," presented at the Night Photo Summit 2023, Feb. 4, 2023. Available: `https://2023.nightphotosummit.com/talks/sprites-elves-ghosts-more-photographing-transient-luminous-events/`. Accessed: Apr. 14, 2025.

[32] R. Nemiroff and J. Bonnell, "Sprites from space," *Astronomy Picture of the Day*, NASA, Aug. 21, 2015. Available: `https://apod.nasa.gov/apod/ap150821.html`. Accessed: Apr. 14, 2025.

[33] European Space Agency, "Elves seen from space," Jan. 20, 2021. Available: `https://www.esa.int/ESA_Multimedia/Videos/2021/01/Elves_seen_from_space`. Accessed: Apr. 14, 2025.

support research aimed at improving aurora forecasting, separating auroral emissions from nighttime lights, and understanding how geomagnetic disturbances affect GPS, communication, and electrical power grid systems [293], [294]. However, many studies remain limited by the availability of high spatial and temporal resolution datasets.

Satellite observations can provide improved coverage of auroral phenomena and enable measurements across spectral ranges including infrared and ultraviolet that are difficult to observe from ground-based systems due to atmospheric absorption [295].

Nevertheless, auroral emissions evolve rapidly and research primarily depends on detailed spectral measurements rather than spatial cueing. Because observations focus on continuous monitoring of known high-latitude regions, the structural advantage of a Tip and Cue architecture is limited for aurora monitoring.

The application framework indicates that AI-based Tip and Cue is mainly advantageous in situations where three conditions coincide: the target location is initially uncertain, high spatial resolution is needed to verify what has been detected, and the value of the observation decreases rapidly with time. This is particularly visible in applications like military surveillance, illegal maritime activity detection, wildfire monitoring, and post-disaster damage assessment, where a low-resolution wide-swath observation alone is not sufficient, but where waiting for a conventional follow-up can significantly reduce operational value.

In contrast, the benefit is limited when the area of interest is already known in advance, when processes evolve gradually over days to years, or when existing monitoring systems already provide timely and reliable observations. The framework therefore helps distinguish applications in which Tip and Cue offers a real architectural advantage from those in which conventional targeted imaging or continuous single-sensor monitoring is already sufficient.

10

Conclusion

Wildfire spread, illegal activity at sea, and changes in wildlife abundance are still often missed or observed too late by current Earth Observation systems. A major reason is the persistent trade-off between wide-area coverage and high spatial resolution. This thesis therefore investigated AI-Based Tip and Cue as a strategy to overcome this limitation without simply relying on increasingly large satellite constellations. In this concept, a wide-field Tip satellite first detects a potential target, after which a high-resolution Cue satellite performs targeted follow-up through off-nadir pointing.

The work addressed the following research question:

> *How do orbital configuration and off-nadir imaging geometry influence the detection performance and mission feasibility of AI-Based Tip and Cue systems?*

The results show that the detection performance and mission feasibility of AI-Based Tip and Cue depend on the interaction between orbital configuration and off-nadir imaging geometry. A larger allowable off-nadir angle reduces latency, increases viewing time, and improves target reachability, whereas a shorter time delay between Tip and Cue reduces viewing time and target reachability, but improves the off-nadir imaging geometry of the final observation. Coordinated Tip-and-Cue formations achieved the strongest performance in the benchmark compared with traditional VHR constellations and independently operating Tip and Cue satellites. In addition, the study shows that reliable mission evaluation requires representative modelling of off-nadir conditions and target dynamics.

SQ1: Benchmark and Simulation Framework
To reach these results, an integrated benchmark and simulation framework was developed that couples three components: an Orbital Simulation Framework, an Off-Nadir Imaging Rendering Pipeline, and an Onboard Detection Module. Together, these components enable to evaluate AI-Based Tip and Cue as one end-to-end mission chain, from target visibility and tasking to generated Cue imagery and final successful confirmation.

The benchmark was defined around whale detection as a demanding demonstration case. Whales were selected because they are ecologically relevant, spatially small, mobile, sparsely distributed, and only briefly visible at the surface. This makes them well suited to testing the trade-offs inherent in AI-Based Tip and Cue. The main benchmark metric was defined as successful detection capacity, measuring the number of successfully confirmed whales per satellite per simulated day. This was complemented by latency, viewing time, AP_{50}, precision, and recall.

The framework was verified and validated at component level. The orbital simulation was visualized and checked against expected mission geometry. The off-nadir rendering pipeline was compared against actual Φ-Sat-2 reflectance data and showed close agreement, with median deviations in the order of approximately 1-14% depending on dataset and spectral band. The DEIMv2 transformer-based detector was optimized through staged hyperparameter tuning with location-based cross-validation, and then deployed in the integrated simulation chain.

SQ2: Imaging Effects and Off-Nadir Observation Angle
It was demonstrated that radiometric and geometric imaging conditions directly affect Cue detection performance. Models trained on simplified nadir-like data overestimate operational performance once they are deployed on representative off-nadir Cue imagery. Including radiometric realism in the training data proved more important for transfer performance compared to including geometric realism. The best representative model achieved an AP_{50} of 0.757 on the unseen-domain test set, compared to 0.656 when radiometric effects were omitted and 0.434 when both geometric and radiometric effects were omitted. This shows that Cue performance cannot be evaluated realistically without representative off-nadir training and test data, as done in prior literature.

The off-nadir sweep further showed that increasing the maximum allowable Cue off-nadir angle has two opposite effects. On the one hand, a larger off-nadir range increases the fraction of the Tip window that can be reached and therefore increases the number of executable Cue tasks. On the other hand, detection performance declines at larger viewing angles because whales become smaller, more distorted, and more affected by sun glint. In the present benchmark, detector performance typically started to decline beyond approximately $40°$, depending on the effective ground sampling distance of the input data. For high resolution Pelagos2016 data from the Whales in Space Dataset [100], the successful detection capacity increased up to around $40°$, whereas low resolution Auckland2006 data degraded earlier, consistent with the smaller effective whale size in that dataset.

This means that the operationally favorable off-nadir range is determined by the interaction of detector quality and reachability. The most useful regime typically lies where enough of the Tip window becomes reachable while the Cue detector still operates under acceptable off-nadir conditions.

SQ3: Tip-Cue Time Delay vs Mission Performance
The third finding is that Tip-Cue spacing mainly changes the balance between latency, viewing geometry, and the number of reachable tasks. Over the tested delay range, the successful detection capacity remained relatively stable, but this does not mean that Tip-Cue spacing is unrelevant, while different effects compensate each other.

At shorter Tip-Cue delays, fewer tasks can be handled, but those tasks are observed at smaller off-nadir angles and are therefore detected under more favorable conditions. At larger delays, more tasks are geometrically reachable, but the follow-up observations occur under less favorable geometry. This explains why successful detection capacity changed only weakly over the tested range.

For mission design, this means that no universal optimal time delay exists. If the priority is to minimize latency and minimize off-nadir angle, the favorable configuration is to place the Cue satellite behind the Tip with a spacing close to the Tip processing and transmission time, such that the Cue reaches the target close to nadir. If the priority is to handle more targets and increase scheduling flexibility, a larger spacing becomes beneficial because it allows more forward pointing and longer viewing time. Time delay and off-nadir capability therefore have to be chosen together.

SQ4: Tip-and-Cue Formations vs Traditional EO Strategies
The comparative evaluation showed that AI-Based Tip and Cue shows can outperform traditional Earth Observation strategies following the end-to-end benchmark performance. Within the benchmark, a single Tip-and-Cue system with only two satellites achieved 50.7 successful confirmations per simulated day at mission level, compared to 44.0 and 43.3 for the two 16-satellite VHR constellations. On a per-satellite basis, the difference is even larger: 25.3 successful detections per satellite for the single Tip-and-Cue baseline, versus 2.8 and 2.7 for the VHR constellations. This indicates that a two-satellite Tip-and-Cue formation architecture can approach or exceed the useful output of a significantly larger VHR-only system, potentially reducing the number of satellites by 87.5%.

An actual Tip and Cue case, with 12 independently operating Tip and Cue satellites based on Sentinel-2, Landsat, and WorldView legion orbital geometry, achieved lower successful detection capacity per satellite compared to a 16-satellite Tip-and-Cue formation constellation. 14.0 versus 20.6, while the mean latency increased from 5.0 minutes to 58.6 minutes. This shows that AI-Based Tip and Cue becomes more effective when Tip detection and Cue follow-up are temporally coordinated. In the independent case, the large latency caused a substantial loss of confirmation opportunities, especially because the targets were moving of Cue swath view.

SQ5: Application Framework
To extend the work beyond the whale detection use case, an applications framework was developed to evaluate where AI-Based Tip and Cue is expected to provide operational benefit. Over 47 candidate Earth Observation applications were classified into high, moderate, and limited expected benefit. The high-benefit group included applications in which rapid targeted follow-up, uncertain target location, and high spatial detail coincide, including wildfire response, illegal activity detection, wildlife monitoring, and rip current alerts.

Implications and Future Outlook
Overall, this thesis lays the foundation for AI-Based Tip and Cue as a researchable and testable Earth Observation mission concept. It showed that mission geometry, off-nadir image formation, and onboard detection performance have to be evaluated together.

The thesis makes four main contributions:

- It delivers an end-to-end mission-analysis framework for AI-Based Tip and Cue by coupling orbital simulation, off-nadir image rendering, and onboard detection in a single evaluation chain.
- It defines a benchmark methodology based on successful detection capacity, latency, viewing time, and detector performance, enabling consistent comparison between Tip-and-Cue architectures and traditional Earth Observation systems.
- It establishes a physically grounded off-nadir dataset-generation pipeline that transforms limited annotated VHR input data into mission-relevant training and evaluation data by simulating acquisition geometry and radiometric conditions, expanding 633 input patches to 13,925 distinct images in the present setup.
- It demonstrates through the whale-monitoring benchmark that AI-Based Tip and Cue performance is governed jointly by accessibility, latency, viewing time and off-nadir detection, and that a two-satellite Tip-and-Cue formation can outperform conventional VHR constellations under the tested conditions.

At the same time, several assumptions and limitations define the scope of the reported results. A realistic Tip detector was not yet included, because suitable Tip data were not available for the whale use case. The rendering pipeline is closer to a bottom-of-atmosphere representation than a full top-of-atmosphere sensor chain, because the outgoing atmospheric path was not modelled. Whale submersion was not included, which likely makes long-delay confirmation results optimistic. The detector was evaluated on one-class whale detection and on cropped patches, so performance in full-scene multi-class conditions remains to be demonstrated.

Future work should therefore focus on integrating Tip-side imagery and detection, improving the realism of the off-nadir rendering pipeline, moving from cropped patches to larger or full-scene inputs, and extending the framework to multi-class and additional application domains. Further development should also include higher-fidelity operational constraints including communication, downlink, power, and onboard prioritization.

In conclusion, this thesis showed that AI-Based Tip and Cue can improve Earth Observation capability without simply relying on ever-larger constellations. It demonstrated that orbital configuration and off-nadir imaging geometry influence mission feasibility through one coupled chain from reachability and latency to image quality and final successful confirmation. Under the benchmark conditions used here, coordinated Tip-and-Cue formations showed strong potential to outperform traditional architectures while using fewer satellites. For Earth Observation problems that require rapid targeted follow-up under limited resources, this potential should be considered explicitly in future mission design.

References

[1] K. Abbass, M. Z. Qasim, H. Song, M. Murshed, H. Mahmood, and I. Younis, "A review of the global climate change impacts, adaptation, and sustainable mitigation measures," *Environmental Science and Pollution Research International*, vol. 29, no. 28, pp. 42 539–42 559, 2022. DOI: 10.1007/s11356-022-19718-6

[2] S. Seitzinger et al., "Relevance of earth observations of essential climate variables in wildfire adaptation," *Remote Sensing of Environment*, vol. 332, p. 115 082, 2026, ISSN: 0034-4257. DOI: 10.1016/j.rse.2025.115082 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425725004869

[3] D. J. Agnew et al., "Estimating the worldwide extent of illegal fishing," *PLOS ONE*, vol. 4, no. 2, pp. 1–8, Feb. 2009. DOI: 10.1371/journal.pone.0004570

[4] N. I. Fawzi, V. N. Husna, and J. A. Helms, "Measuring deforestation using remote sensing and its implication for conservation in gunung palung national park, west kalimantan, indonesia," *IOP Conference Series: Earth and Environmental Science*, vol. 149, p. 012 038, May 2018, ISSN: 1755-1315. DOI: 10.1088/1755-1315/149/1/012038

[5] S. Battersby, M. Hodgson, and J. Wang, "Spatial resolution imagery requirements for identifying structure damage in a hurricane disaster: A cognitive approach," *Photogrammetric Engineering and Remote Sensing*, vol. 78, pp. 625–635, Jun. 2012. DOI: 10.14358/PERS.78.6.625

[6] D. S. Matheson and P. E. Dennison, "Evaluating the effects of spatial resolution on hyperspectral fire detection and temperature retrieval," *Remote Sensing of Environment*, vol. 124, pp. 780–792, 2012, ISSN: 0034-4257. DOI: 10.1016/j.rse.2012.06.026 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425712002623

[7] S. Pandey et al., "Daily detection and quantification of methane leaks using sentinel-3: A tiered satellite observation approach with sentinel-2 and sentinel-5p," *Remote Sensing of Environment*, vol. 296, p. 113 716, 2023, ISSN: 0034-4257. DOI: 10.1016/j.rse.2023.113716 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425723002675

[8] J. Fan, M. Guo, L. Zhang, J. Liu, and Y. Li, "A marine ship detection method for super-resolution sar images based on hierarchical multi-scale mask r-cnn," *Frontiers in Marine Science*, vol. 12, 2025, ISSN: 2296-7745. DOI: 10.3389/fmars.2025.1574991 [Online]. Available: https://www.frontiersin.org/journals/marine-science/articles/10.3389/fmars.2025.1574991

[9] M.-M. Quaggiotto et al., "Past, present and future of the ecosystem services provided by cetacean carcasses," *Ecosystem Services*, vol. 54, p. 101 406, 2022, ISSN: 2212-0416. DOI: 10.1016/j.ecoser.2022.101406

[10] D. Cook, L. Malinauskaite, B. Davíðsdóttir, H. Ögmundardóttir, and J. Roman, "Reflections on the ecosystem services of whales and valuing their contribution to human well-being," *Ocean and Coastal Management*, vol. 186, p. 105 100, 2020, ISSN: 0964-5691. DOI: 10.1016/j.ocecoaman.2020.105100 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0964569120300107

[11] E. Guirado, S. Tabik, M. L. Rivas, D. Alcaraz-Segura, and F. Herrera, "Whale counting in satellite and aerial images with deep learning," *Scientific Reports*, vol. 9, no. 1, p. 14 259, 2019, ISSN: 2045-2322. DOI: 10.1038/s41598-019-50795-9

[12] H. Frouin-Mouy et al., "Beaked whale dive behavior and acoustic detection range off louisiana using three-dimensional acoustic tracking," *PLOS ONE*, vol. 21, no. 2, pp. 1–28, Feb. 2026. DOI: 10.1371/journal.pone.0340398

[13] S. L. WATWOOD, P. J. O. MILLER, M. JOHNSON, P. T. MADSEN, and P. L. TYACK, "Deep-diving foraging behaviour of sperm whales (physeter macrocephalus)," *Journal of Animal Ecology*, vol. 75, no. 3, pp. 814–825, 2006. DOI: 10.1111/j.1365-2656.2006.01101.x eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-2656.2006.01101.x. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2656.2006.01101.x

[14] D. Phiri, M. Simwanda, S. Salekin, V. R. Nyirenda, Y. Murayama, and M. Ranagalage, "Sentinel-2 data for land cover/use mapping: A review," *Remote Sensing*, vol. 12, no. 14, 2020, ISSN: 2072-4292. DOI: 10.3390/rs12142291 [Online]. Available: https://www.mdpi.com/2072-4292/12/14/2291

[15] S. J. Cantrell et al., "System characterization report on the worldview-3 imager," U.S. Geological Survey, Reston, VA, Open-File Report 2021–1030I, version Version 1.1, 2021, p. 40. DOI: 10.3133/ofr20211030I [Online]. Available: https://pubs.usgs.gov/publication/ofr20211030I

[16] C. Chen and W. Yang, "The impact of large constellations on space debris environment and its countermeasures," in *Proceedings of the 8th European Conference for Aeronautics and Space Sciences (EUCASS)*, Madrid, Spain: EUCASS Association, 2019. DOI: 10.13009/EUCASS2019-885

[17] eoPortal. "Jilin / gaofen constellation." Last updated 2025-03-03; accessed 2026-02-24. [Online]. Available: https://www.eoportal.org/satellite-missions/jilin-con#jilin-constellation

[18] K. Nock, K. Aaron, and D. McKnight, "Removing orbital debris with less risk," *Journal of Spacecraft and Rockets*, vol. 50, pp. 365–379, Mar. 2013. DOI: 10.2514/1.A32286

[19] W. L. Filho, I. R. Abubakar, J. D. Hunt, and M. A. P. Dinis, "Managing space debris: Risks, mitigation measures, and sustainability challenges," *Sustainable Futures*, vol. 10, p. 100 849, 2025, ISSN: 2666-1888. DOI: 10.1016/j.sftr.2025.100849 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666188825004149

[20] J. Mackintosh, K. Smith, and C. McGrath, "Collision risk from performance requirements in earth observation mission design," *Advances in Space Research*, vol. 77, no. 5, pp. 5941–5961, 2026, ISSN: 0273-1177. DOI: 10.1016/j.asr.2026.01.019 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0273117726000438

[21] S. Vellas and M. Bernou, "IDEA I-2022-00382 - Dual Camera Satellite with On-Board AI-Based Decision-Making Capabilities," OHB-Hellas (Prime), FORTH (Subcontractor), Germany, Executive Summary Report 4000138071, Dec. 2022, Prepared for the ESA Cognitive Cloud Computing in Space campaign. [Online]. Available: https://nebula.esa.int/sites/default/files/neb_study/2675/C4000138110ExS.pdf

[22] A. Alzubairi, A. Tameem, and B. Kada, "Spacecraft formation flying orbital control for earth observation mission," *Scientific African*, vol. 26, e02391, 2024, ISSN: 2468-2276. DOI: 10.1016/j.sciaf.2024.e02391

[23] B. Cudzilo, K. Foley, and C. Smith, "The ability of a small satellite constellation to tip and cue other commercial assets," in *26th Annual AIAA/USU Conference on Small Satellites*, SSC12-IV-5, AIAA/USU, Logan, Utah, USA, 2012. [Online]. Available: https://digitalcommons.usu.edu/smallsat/2012/all2012/33

[24] M. I. Ali. "Tip and cue technique for satellite monitoring of moving objects," Accessed: Dec. 31, 2024. [Online]. Available: https://www.iceye.com/blog/tip-and-cue-technique-for-efficient-near-real-time-satellite-monitoring-of-moving-objects

[25] C. R. Post, "Towards automation of tipping and cueing between small satellites in a constellation," AFIT-ENG-MS-17-M-061. Approved for public release; distribution unlimited., Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, Mar. 2017.

[26] V. Rúžička, G. Mateo-Garcia, A. Vaughan, L. Guanter, L. Gómez-Chova, and J. Parr, "STARCOP: Automated and self-improving follow-up verification of detrimental human-activity from LEO," European Space Agency (ESA), Technical Report Starcop 1-2022-00380, Nov. 2022, Published as part of the ESA Cognitive Cloud Computing in Space Campaign. Implemented as ESA Initial Support for Innovation. Agreement/PO number: 4000138110. [Online]. Available: https://nebula.esa.int/sites/default/files/neb_study/2675/C4000138110ExS.pdf

[27] G. Weissman, A. Ivry, and I. Cohen, *An automated tip-and-cue framework for optimized satellite tasking and visual intelligence*, Dec. 2025. arXiv: 2512.09670 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2512.09670

[28] M. Stephenson and H. Schaub, "Autonomous tip-and-cue earth-observing constellation tasking with reinforcement learning," in *Proceedings of the IEEE Aerospace Conference*, 979-8-3315-7360-7/26/$31.00 ©2026 IEEE, IEEE, 2026.

[29] A. Saleh, M. A. Zulkifley, H. H. Harun, F. Gaudreault, I. Davison, and M. Spraggon, "Forest fire surveillance systems: A review of deep learning methods," English, *Heliyon*, vol. 10, no. 1, e23127, Jan. 2024, ISSN: 2405-8440. DOI: 10.1016/j.heliyon.2023.e23127

[30] Z. Li, S. Xu, and Q. Weng, "Beyond clouds: Seamless flood mapping using harmonized landsat and sentinel-2 time series imagery and water occurrence data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 216, pp. 185–199, 2024, ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2024.07.022

[31] P. Berezina and D. Liu, "Hurricane damage assessment using coupled convolutional neural networks: A case study of hurricane michael," *Geomatics, Natural Hazards and Risk*, vol. 13, no. 1, pp. 414–431, 2022. DOI: 10.1080/19475705.2022.2030414

[32] S. Majumdar, "The role of remote sensing and gis in military strategy to prevent terror attacks," in *Intelligent Data Analytics for Terror Threat Prediction: Architectures, Methodologies, Techniques, and Applications*. 2021, pp. 79–94. DOI: 10.1002/9781119711629.ch4

[33] T. E. Skosana, K. J. Esler, and A. J. Rebelo, "Exploring the trade-offs between spatial and spectral resolution in mapping invasive alien trees," *Ecological Informatics*, vol. 92, p. 103 448, 2025, ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2025.103448 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574954125004571

[34] European Space Agency (ESA), *Planetscope*, https://earth.esa.int/eogateway/missions/planetscope, No date. Accessed: 2026-02-24.

[35] M. Sias, *Muon space releases first light images from firesat protoflight*, https://www.muonspace.com/press/muon-space-releases-first-light-images-from-firesat-protoflight, Press release, accessed 2026-02-24, Jun. 2025.

[36] Novaspace Intelligence Hub, *Earth Observation Satellite Systems, 18th edition*, https://nova.space/hub/product/earth-observation-satellite-systems-database/, Industry report on global Earth observation satellite systems and market outlook, accessed 2026-02-24, 2025.

[37] Windward, *Utilizing tipping and cueing for enhanced visibility*, 2025. Accessed: Apr. 3, 2025. [Online]. Available: https://windward.ai/knowledge-base/how-to-employ-tipping-and-cueing-to-increase-visibility-optimize-decision-making/

[38] Z. Hajkova. "Understanding ona in satellite imagery: What is off nadir angle and what is it used for?" European Space Imaging, Accessed: Mar. 6, 2026. [Online]. Available: https://www.euspaceimaging.com/blog/2024/01/31/what-is-ona-off-nadir-angle-in-satellite-imagery/

[39] N. Lazreg, R. Bouchiha, and K. Besbes, "Registration and correction techniques in cubesat remote sensing images," May 2017, pp. 1–7. DOI: 10.1109/ICEMIS.2017.8273059

[40] Chang Guang Satellite Technology Co., Ltd. "Off-nadir imaging by optical remote sensing satellites," Chang Guang Satellite Technology Co., Ltd., Accessed: Mar. 6, 2026. [Online]. Available: https://www.jl1global.com/offnadir-imaging-by-optical-remote-sensing-satellites.html

[41] F. Peng, M. Guo, H. Hu, T. Yan, and L. Jiang, "Off-nadir satellite image scene classification: Benchmark dataset, angle-aware active domain adaptation, and angular impact analysis," *Remote Sensing*, vol. 17, no. 22, 2025, ISSN: 2072-4292. DOI: 10.3390/rs17223697 [Online]. Available: https://www.mdpi.com/2072-4292/17/22/3697

[42] A. Delplanque, J. Théau, S. Foucher, G. Serati, S. Durand, and P. Lejeune, "Wildlife detection, counting and survey using satellite imagery: Are we there yet?" *GIScience & Remote Sensing*, vol. 61, no. 1, 2024. DOI: 10.1080/15481603.2024.2348863

[43] S. Kapoor, M. Kumar, and M. Kaushal, "Deep learning based whale detection from satellite imagery," *Sustainable Computing: Informatics and Systems*, vol. 38, p. 100 858, 2023, ISSN: 2210-5379. DOI: 10.1016/j.suscom.2023.100858

[44] C. B. Khan et al., "A biologist's guide to the galaxy: Leveraging artificial intelligence and very high-resolution satellite imagery to monitor marine mammals from space," *Journal of Marine Science and Engineering*, vol. 11, no. 3, 2023, ISSN: 2077-1312. DOI: 10.3390/jmse11030595

[45] R. Montes and C. Ureña, "An overview of brdf models," Dept. Lenguajes y Sistemas Informáticos, University of Granada, Granada, Spain, Technical Report LSI-2012-001, 2012.

[46] J. A. Shaw and M. Vollmer, "Blue sun glints on water viewed through a polarizer," *Appl. Opt.*, vol. 56, no. 19, G36–G41, Jul. 2017. DOI: 10.1364/AO.56.000G36 [Online]. Available: https://opg.optica.org/ao/abstract.cfm?URI=ao-56-19-G36

[47] J. Hostetler and H. Cowardin, "Experimentally-derived bidirectional reflectance distribution function data in support of the orbital debris program office," NASA Johnson Space Center, Orbital Debris Program Office, Conference Paper IOC 6148, 2019, Presented at the International Orbital Debris Conference. [Online]. Available: https://ntrs.nasa.gov/api/citations/20190030914/downloads/20190030914.pdf

[48] L. Zhang, E. Rupnik, T. D. Nguyen, S. Jacquemoud, and Y. Klinger, "Brdf-nerf: Neural radiance fields with optical satellite images and brdf modelling," *International Journal of Applied Earth Observation and Geoinformation*, vol. 143, p. 104 747, 2025, ISSN: 1569-8432. DOI: 10.1016/j.jag.2025.104747 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569843225003942

[49] A. W. Bailey, E. A. Early, E. Ahmed, A. W. Wharmby, and S. S. Kumru, "Probabilistic water bidirectional reflectance distribution function, version 2," Air Force Research Laboratory, 711th Human Performance Wing, Airman Systems Directorate, Bioeffects Division, Optical Radiation Bioeffects Branch, JBSA Fort Sam Houston, Texas, Tech. Rep. AFRL-RH-FS-TR-2023-0002, Jan. 2023, Interim Technical Report. Distribution A: Approved for public release; distribution is unlimited.

[50] D. D'Alimonte et al., "Comparison of correction methods for bidirectional effects in ocean colour remote sensing," *Remote Sensing of Environment*, vol. 321, p. 114 606, 2025, ISSN: 0034-4257. DOI: 10.1016/j.rse.2025.114606 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425725000100

[51] S. D. Butler, "Experimental and theoretical basis for a closed-form spectral brdf model," DTIC Accession Number: ADA626700. Distribution Statement A: Approved for public release; distribution unlimited., PhD Dissertation, Graduate School of Engineering and Management, Wright-Patterson Air Force Base, Ohio, Sep. 2015. [Online]. Available: https://apps.dtic.mil/sti/tr/pdf/ADA626700.pdf

[52] K. Miller and A. Lohn, "Onboard ai: Constraints and limitations," Center for Security and Emerging Technology, Aug. 2023. DOI: 10.51593/2022CA008

[53] A. Silva et al., "Resource-constrained onboard inference of 3d object detection and localisation in point clouds targeting self-driving applications," *Sensors*, vol. 21, no. 23, 2021, ISSN: 1424-8220. DOI: 10.3390/s21237933 [Online]. Available: https://www.mdpi.com/1424-8220/21/23/7933

[54] A. Dorise, M. Bellizzi, A. Girard, B. Francesconi, and S. May, *Explaining raw data complexity to improve satellite onboard processing*, 2025. arXiv: 2510.06858 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2510.06858

[55] I. Purwono et al., "Understanding of convolutional neural network (cnn): A review," *International Journal of Robotics and Control Systems*, vol. 2, pp. 739–748, Jan. 2023. DOI: 10.31763/ijrcs.v2i4.888

[56] D. Hindustani, S. Hindustani, and P. Nguyen, *Tackling tuberculosis: A comparative dive into machine learning for tuberculosis detection*, Dec. 2025. DOI: 10.48550/arXiv.2512.02364

[57] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, p. 99, 2024. DOI: 10.1007/s10462-024-10721-6

[58] K. O'Shea and R. Nash, *An introduction to convolutional neural networks*, 2015. arXiv: 1511.08458 [cs.NE]. [Online]. Available: https://arxiv.org/abs/1511.08458

[59] A. Dosovitskiy et al., *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. arXiv: 2010.11929 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2010.11929

[60] C. Kameswari et al., "An overview of vision transformers for image processing: A survey," *International Journal of Advanced Computer Science and Applications*, vol. 14, Jan. 2023. DOI: 10.14569/IJACSA.2023.0140830

[61] S. Saha and L. Xu, "Vision transformers on the edge: A comprehensive survey of model compression and acceleration strategies," *Neurocomputing*, vol. 643, p. 130 417, 2025, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2025.130417 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231225010896

[62] A. Vaswani et al., *Attention is all you need*, 2023. arXiv: 1706.03762 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1706.03762

[63] J. Maurício, I. Domingues, and J. Bernardino, "Comparing vision transformers and convolutional neural networks for image classification: A literature review," *Applied Sciences*, vol. 13, no. 9, 2023, ISSN: 2076-3417. DOI: 10.3390/app13095521 [Online]. Available: https://www.mdpi.com/2076-3417/13/9/5521

[64] S. Oh, N. Kim, and J. Ryu, "Analyzing to discover origins of cnns and vit architectures in medical images," *Scientific Reports*, vol. 14, no. 1, p. 8755, 2024. DOI: 10.1038/s41598-024-58382-3

[65] Y. Haruna, S. Qin, A. H. Adama Chukkol, A. A. Yusuf, I. Bello, and A. Lawan, "Exploring the synergies of hybrid convolutional neural network and vision transformer architectures for computer vision: A survey," *Engineering Applications of Artificial Intelligence*, vol. 144, p. 110 057, 2025, ISSN: 0952-1976. DOI: 10.1016/j.engappai.2025.110057 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197625000570

[66] A. Khan et al., "A survey of the vision transformers and their cnn-transformer based variants," *Artificial Intelligence Review*, vol. 56, no. S3, pp. 2917–2970, Oct. 2023, ISSN: 1573-7462. DOI: 10.1007/s10462-023-10595-0

[67] M. Kawai, N. Ota, and S. Yamaoka, *Large-scale pretraining on pathological images for fine-tuning of small pathological benchmarks*, 2023. arXiv: 2303.15693 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2303.15693

[68] K. Sun and M. Dredze, "Amuro & char: Analyzing the relationship between pre-training and fine-tuning of large language models," version v2, *arXiv preprint arXiv:2408.06663*, Aug. 2024. DOI: 10.48550/arXiv.2408.06663 [Online]. Available: https://arxiv.org/abs/2408.06663

[69] C. Ramani, "Transfer learning: Backbone selection and fine-tuning for binary image classification," Jan. 2026. DOI: 10.36227/techrxiv.176761849.95775411/v1

[70] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. arXiv: 1512.03385 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1512.03385

[71] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, *Rethinking the inception architecture for computer vision*, 2015. arXiv: 1512.00567 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1512.00567

[72] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, *Densely connected convolutional networks*, 2018. arXiv: 1608.06993 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1608.06993

[73] M. Tan and Q. V. Le, *Efficientnet: Rethinking model scaling for convolutional neural networks*, 2020. arXiv: 1905.11946 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1905.11946

[74] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," *CoRR*, vol. abs/2012.12877, 2020. arXiv: 2012.12877. [Online]. Available: https://arxiv.org/abs/2012.12877

[75] D. Zhou et al., *Deepvit: Towards deeper vision transformer*, 2021. arXiv: 2103.11886 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2103.11886

[76] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 10 012–10 022.

[77] O. Elharrouss et al., "Vits as backbones: Leveraging vision transformers for feature extraction," *Information Fusion*, vol. 118, p. 102 951, 2025, ISSN: 1566-2535. DOI: 10.1016/j.inffus.2025.102951 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253525000247

[78] S. Ren, K. He, R. Girshick, and J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, 2016. arXiv: 1506.01497 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1506.01497

[79] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *You only look once: Unified, real-time object detection*, 2016. arXiv: 1506.02640 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1506.02640

[80] M. Hussain, *Yolov5, yolov8 and yolov10: The go-to detectors for real-time vision*, 2024. arXiv: 2407.02988 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2407.02988

[81] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Cham: Springer International Publishing, 2020, pp. 213–229, ISBN: 978-3-030-58452-8.

[82] H. Zhang et al., *Dino: Detr with improved denoising anchor boxes for end-to-end object detection*, 2022. arXiv: 2203.03605 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2203.03605

[83] S. Wang, C. Xia, F. Lv, and Y. Shi, *Rt-detrv3: Real-time end-to-end object detection with hierarchical dense positive supervision*, 2024. arXiv: 2409.08475 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2409.08475

[84] O. Ronneberger, P. Fischer, and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, 2015. arXiv: 1505.04597 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1505.04597

[85] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, *Unet++: A nested u-net architecture for medical image segmentation*, 2018. arXiv: 1807.10165 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1807.10165

[86] K. He, G. Gkioxari, P. Dollár, and R. Girshick, *Mask r-cnn*, 2018. arXiv: 1703.06870 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1703.06870

[87] J. Stodt, C. Reich, and N. Clarke, "Unified intersection over union for explainable artificial intelligence," in Apr. 2024, pp. 758–770, ISBN: 978-3-031-47723-2. DOI: 10.1007/978-3-031-47724-9_50

[88] P. T. Fretwell, I. J. Staniland, and J. Forcada, "Whales from space: Counting southern right whales by satellite," *PLOS ONE*, vol. 9, no. 2, pp. 1–9, Feb. 2014. DOI: 10.1371/journal.pone.0088655

[89] A. Borowicz et al., "Aerial-trained deep learning networks for surveying cetaceans from satellite imagery," *PLOS ONE*, vol. 14, no. 10, pp. 1–15, Oct. 2019. DOI: 10.1371/journal.pone.0212532

[90] C. Höschle et al., "Satellite surveys prove a reliable monitoring method for high latitude southern right whale habitat," International Whaling Commission, Tech. Rep., Apr. 2022. [Online]. Available: https://www.spacewhales.de/wp-content/uploads/2022/06/SC_68D_SH_12.pdf

[91] K. Green et al., "Gray whale detection in satellite imagery using deep learning," *Remote Sensing in Ecology and Conservation*, vol. 9, Jun. 2023. DOI: 10.1002/rse2.352

[92] P. C. Gray, K. C. Bierlich, S. A. Mantell, A. S. Friedlaender, J. A. Goldbogen, and D. W. Johnston, "Drones and convolutional neural networks facilitate automated and accurate cetacean species identification and photogrammetry," *Methods in Ecology and Evolution*, vol. 10, no. 9, pp. 1490–1500, 2019. DOI: 10.1111/2041-210X.13246

[93] J. Boulent et al., "Scaling whale monitoring using deep learning: A human-in-the-loop solution for analyzing aerial datasets," *Frontiers in Marine Science*, vol. 10, 2023, ISSN: 2296-7745. DOI: 10.3389/fmars.2023.1099479

[94] M. Gheibi, "Helping biologists find whales: Ai-in-the-loop support for environmental dataset creation," M.S. thesis, Dalhousie University, Halifax, Nova Scotia, Nov. 2021. [Online]. Available: http://hdl.handle.net/10222/81075

[95] P. Q. Lee, K. Radhakrishnan, D. A. Clausi, K. A. Scott, L. Xu, and M. M. and, "Beluga whale detection in the cumberland sound bay using convolutional neural networks," *Canadian Journal of Remote Sensing*, vol. 47, no. 2, pp. 276–294, 2021. DOI: 10.1080/07038992.2021.1901221

[96] S. Vasavi, V. Sripathi, and C. Mouli Simma, "Visualization of humpback whale tracking on edge device using space-borne remote sensing data for indian ocean," *The Egyptian Journal of Remote Sensing and Space Sciences*, vol. 27, no. 4, pp. 705–715, 2024, ISSN: 1110-9823. DOI: 10.1016/j.ejrs.2024.10.004

[97] S. Vasavi, S. Akshaya, and C. Pranavi, "Advanced remote sensing techniques for underwater fin whale detection in the indian ocean," in *2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN)*, 2024, pp. 1328–1334. DOI: 10.1109/CICN63059.2024.10847459

[98] C. C. G. Bamford et al., "A comparison of baleen whale density estimates derived from overlapping satellite imagery and a shipborne survey," *Scientific Reports*, vol. 10, no. 1, p. 12 985, 2020, ISSN: 2045-2322. DOI: 10.1038/s41598-020-69887-y

[99] A. M. Belanger, B. A. H. Sherbo, J. D. Roth, and C. A. Watt, "Use of satellite imagery to estimate distribution and abundance of cumberland sound beluga whales reveals frequent use of a glacial river estuary," *Frontiers in Marine Science*, vol. 10, 2024, ISSN: 2296-7745. DOI: 10.3389/fmars.2023.1305536

[100] H. C. Cubaynes and P. T. Fretwell, "Whales from space dataset, an annotated satellite image dataset of whales for training machine learning models," *Scientific Data*, vol. 9, no. 1, p. 245, 2022, ISSN: 2052-4463. DOI: 10.1038/s41597-022-01377-4

[101] A. Gaur, C. Liu, X. Lin, N. Karapetyan, and Y. Aloimonos, "Whale detection enhancement through synthetic satellite images," in *OCEANS 2023 - MTS/IEEE U.S. Gulf Coast*, IEEE, Sep. 2023, pp. 1–7. DOI: 10.23919/oceans52994.2023.10337400

[102] C. Robinson et al., *Where are the whales: A human-in-the-loop detection method for identifying whales in high-resolution satellite imagery*, 2025. arXiv: 2510.14709 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2510.14709

[103] C. Höschle, H. C. Cubaynes, P. J. Clarke, G. Humphries, and A. Borowicz, "The potential of satellite imagery for surveying whales," *Sensors*, vol. 21, no. 3, 2021, ISSN: 1424-8220. DOI: 10.3390/s21030963

[104] P. T. Fretwell et al., "Using remote sensing to detect whale strandings in remote areas: The case of sei whales mass mortality in chilean patagonia," *PLOS ONE*, vol. 14, no. 10, pp. 1–17, Oct. 2019. DOI: 10.1371/journal.pone.0222498

[105] H. van Deventer, A. Linström, L. Naidoo, N. Job, E. Sieben, and M. Cho, "Comparison between sentinel-2 and worldview-3 sensors in mapping wetland vegetation communities of the grassland biome of south africa, for monitoring under climate change," *Remote Sensing Applications: Society and Environment*, vol. 28, p. 100 875, 2022, ISSN: 2352-9385. DOI: 10.1016/j.rsase.2022.100875 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352938522001835

[106] J. Wu, L. Lin, T. Li, Q. Cheng, C. Zhang, and H. Shen, "Fusing landsat 8 and sentinel-2 data for 10-m dense time-series imagery using a degradation-term constrained deep network," *International Journal of Applied Earth Observation and Geoinformation*, vol. 108, p. 102 738, 2022, ISSN: 1569-8432. DOI: 10.1016/j.jag.2022.102738 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0303243422000642

[107] NASA Jet Propulsion Laboratory. "How nasa is testing ai to make earth-observing satellites smarter." Accessed: 2026-02-17, Jet Propulsion Laboratory, California Institute of Technology. [Online]. Available: https://www.jpl.nasa.gov/news/how-nasa-is-testing-ai-to-make-earth-observing-satellites-smarter

[108] P. Wessel and W. H. F. Smith, "A global self-consistent, hierarchical, high-resolution shoreline database," *Journal of Geophysical Research: Solid Earth*, vol. 101, no. B4, pp. 8741–8743, 1996. DOI: 10.1029/96JB00104

[109] P. Wessel and W. H. F. Smith, *Gshhg: A global self-consistent, hierarchical, high-resolution geography database, version 2.3.7*, Developed at SOEST, University of Hawai'i and NOAA Geosciences Lab. Accessed: 2026-02-17, Jun. 2017. [Online]. Available: https://www.soest.hawaii.edu/pwessel/gshhg/

[110] S. V. Calderan et al., "Surfacing rates, swim speeds, and patterns of movement of antarctic blue whales," *Frontiers in Marine Science*, vol. Volume 10 - 2023, 2023, ISSN: 2296-7745. DOI: 10.3389/fmars.2023.1087967 [Online]. Available: https://www.frontiersin.org/journals/marine-science/articles/10.3389/fmars.2023.1087967

[111] H. Kela et al., "Assessment of humpback whale swimming speeds in two eastern australian bays," *Marine and Freshwater Research*, vol. 75, Nov. 2024. DOI: 10.1071/MF24116

[112] K. Knight, "Whales no faster than other ocean inhabitants despite size," *Journal of Experimental Biology*, vol. 222, no. 20, jeb216044, Oct. 2019, ISSN: 0022-0949. DOI: 10.1242/jeb.216044 eprint: https://journals.biologists.com/jeb/article-pdf/222/20/jeb216044/1978205/jeb216044.pdf.

[113] C. Veness, *Latitude/longitude spherical geodesy tools*, Accessed: 2026-03-01, 2022. [Online]. Available: https://www.movable-type.co.uk/scripts/latlong.html

[114] L. J. S. Pacheco, L. B. T. Santos, L. M. B. Silva, Y. N. Razoumny, N. B. Lima, and A. F. B. A. Prado, "A comparative analysis of walker delta and star satellite constellations for earth remote sensing," *Mathematics in Engineering, Science and Aerospace (MESA)*, vol. 16, no. 2, 2025.

[115] M. Piraux, *A code to describe satellite constellations*, Internet-Draft, IRTF, Systems and Protocol Aspects for Circumstellar Environments RG, Work in progress. Expires 23 April 2026, Oct. 2025. [Online]. Available: https://datatracker.ietf.org/doc/draft-piraux-space-constellation-code/

[116] I. Leyva-Mayorga et al., *Ngso constellation design for global connectivity*, Mar. 2022. DOI: 10.48550/arXiv.2203.16597

[117] S. Jeon and S.-Y. Park, *Communication constellation design of minimum number of satellites with continuous coverage and inter-satellite link*, 2025. arXiv: 2410.03354 [physics.space-ph]. [Online]. Available: https://arxiv.org/abs/2410.03354

[118] P. Gómez, J. Östman, V. M. Shreenath, and G. Meoni, "PAseos Simulates the Environment for Operating multiple Spacecraft," *arXiv:2302.02659 [cs.DC]*, 2023.

[119] L. Maisonobe et al., *CS-SI/Orekit: 13.1.4*, version 13.1.4, Feb. 2026. DOI: 10.5281/zenodo.18527188

[120] N. Bloise, E. Capello, M. Dentis, and E. Punta, "Obstacle avoidance with potential field applied to a rendezvous maneuver," *Applied Sciences*, vol. 7, p. 1042, Oct. 2017. DOI: 10.3390/app7101042

[121] J. Hammerl and H. Schaub, "Orbiting spacecraft relative motion in the inertial frame," *The Journal of the Astronautical Sciences*, vol. 73, Feb. 2026. DOI: 10.1007/s40295-025-00556-w

[122] M. M. Peet, *Lecture 9: 6dof equations of motion*, https://control.asu.edu/Classes/MMAE441/Aircraft/441Lecture9.pdf, No date. Aircraft Dynamics, Illinois Institute of Technology.

[123] L. R. M. Maas, "On the surface area of an ellipsoid and related integrals of elliptic integrals," *Journal of Computational and Applied Mathematics*, vol. 51, no. 2, pp. 237–249, 1994. DOI: 10.1016/0377-0427(92)00009-X

[124] C. R. O. Longo and S. L. Rickman, "Method for the calculation of spacecraft umbra and penumbra shadow terminator points," National Aeronautics and Space Administration, Lyndon B. Johnson Space Center, Houston, Texas, Tech. Rep. NASA Technical Paper 3547, Apr. 1995. [Online]. Available: https://ntrs.nasa.gov/api/citations/19950023025/downloads/19950023025.pdf

[125] M. I. of Technology, *2.004 dynamics and control ii: Lecture 21*, Lecture notes, Massachusetts Institute of Technology, Spring 2008, MIT OpenCourseWare, Mar. 2008. [Online]. Available: https://ocw.mit.edu/courses/mechanical-engineering/2-004-dynamics-and-control-ii-spring-2008/8e6878353de8beaeacbab817cb8c314b_lecture_21.pdf

[126] J. D. Hedley and C. D. Mobley, *Hydrolight 6.0 and ecolight 6.0: Technical documentation*, Version 6.0, 7 January 2021, Numerical Optics Ltd., Tiverton, UK, Jan. 2021. [Online]. Available: https://www.numericaloptics.com/doc/HE60TechDoc.pdf

[127] EUMETSAT, *Brdf correction for sentinel-3 olci water reflectance products*, https://www.eumetsat.int/brdf-correction-s3-olci-water-reflectance-products, Accessed: 2026-02-12, 2024.

[128] D. Harris and A. V. Niekerk, "Radiometric homogenisation of aerial images by calibrating with satellite data," *International Journal of Remote Sensing*, vol. 40, no. 7, pp. 2623–2647, 2019. DOI: 10.1080/01431161.2018.1528404

[129] W. Jakob et al., *Mitsuba 3 renderer*, version 3.1.1, https://mitsuba-renderer.org, 2022.

[130] M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation (3rd ed.)* 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016, p. 1266, ISBN: 9780128006450.

[131] M. Papas, K. de Mesa, and H. W. Jensen, "A physically-based bsdf for modeling the appearance of paper," *Computer Graphics Forum*, vol. 33, no. 4, pp. 133–142, 2014, Proceedings of the 25th Eurographics Symposium on Rendering (EGSR 2014), ISSN: 1467-8659. DOI: 10.1111/cgf.12420

[132] R. Müller and U. Pfeifroth, "Remote sensing of solar surface radiation – a reflection of concepts, applications and input data based on experience with the effective cloud albedo," *Atmospheric Measurement Techniques*, vol. 15, no. 5, pp. 1537–1561, 2022. DOI: 10.5194/amt-15-1537-2022 [Online]. Available: https://amt.copernicus.org/articles/15/1537/2022/

[133] C. Gueymard, "Parameterized transmittance model for direct beam and circumsolar spectral irradiance," *Solar Energy*, vol. 71, no. 5, pp. 325–346, 2001. DOI: 10.1016/S0038-092X(01)00054-8

[134] L. H. Holthuijsen, *Waves in Oceanic and Coastal Waters*. Cambridge: Cambridge University Press, 2007, ISBN: 9780521860284.

[135] D. Lynch, "Visually discerning the curvature of the earth," *Applied Optics*, vol. 47, H39–H43, Jul. 2008. DOI: 10.1364/AO.47.000H39

[136] C. A. Gueymard, "The smarts spectral irradiance model after 25 years: New developments and validation of reference spectra," *Solar Energy*, vol. 187, pp. 233–253, 2019, ISSN: 0038-092X. DOI: 10.1016/j.solener.2019.05.048 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0038092X19305110

[137] S. Ayala Pelaez and C. Deline, *Pysmarts: Smarts python wrapper (simple model of the atmospheric radiative transfer of sunshine)*, Feb. 2020. DOI: 10.11578/dc.20210816.1 [Online]. Available: https://www.osti.gov/biblio/code-62140

[138] C. D. Mobley. "Cox-munk sea surface slope statistics." Ocean Optics Web Book, Creative Commons Attribution License. [Online]. Available: https://www.oceanopticsbook.info/view/surfaces/cox-munk-sea-surface-slope-statistics

[139] M. Kuester, "Radiometric use of worldview-3 imagery," DigitalGlobe, Technical Note, Feb. 2016, Version 2015v2 calibration referenced.

[140] P. Apostolov, "Synthesis of high-selectivity two-dimensional filter banks using sigmoidal function," *Electronics*, vol. 13, no. 21, 2024, ISSN: 2079-9292. DOI: 10.3390/electronics13214146 [Online]. Available: https://www.mdpi.com/2079-9292/13/21/4146

[141] D. R. Bull, "Chapter 4 - digital picture formats and representations," in *Communicating Pictures*, D. R. Bull, Ed., Oxford: Academic Press, 2014, pp. 99–132, ISBN: 978-0-12-405906-1. DOI: 10.1016/B978-0-12-405906-1.00004-0 [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780124059061000040

[142] Roboflow, *Whales from satellite computer vision dataset*, https://universe.roboflow.com/yolov8-whale-detection/whales-from-satellite, Accessed: 2026-03-08, 2025.

[143] L. Zhang, T. Wen, and J. Shi, *Deep image blending*, 2019. arXiv: 1910.11495 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1910.11495

[144] D. P. Zitterbart et al., "Scaling the laws of thermal imaging-based whale detection," *Journal of Atmospheric and Oceanic Technology*, vol. 37, no. 5, pp. 807–824, 2020. DOI: 10.1175/JTECH-D-19-0054.1

[145] C. A. Escobar and D. Restrepo Alvarez, "Estimation of global ocean surface winds blending reanalysis, satellite and buoy datasets," *Remote Sensing Applications: Society and Environment*, vol. 32, p. 101 012, 2023, ISSN: 2352-9385. DOI: 10.1016/j.rsase.2023.101012 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352938523000940

[146] S. Huang, Y. Hou, L. Liu, X. Yu, and X. Shen, *Real-time object detection meets dinov3*, 2026. arXiv: 2509.20787 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2509.20787

[147] R. Sapkota, R. H. Cheppally, A. Sharda, and M. Karkee, *Yolo26: Key architectural enhancements and performance benchmarking for real-time object detection*, 2026. arXiv: 2509.25164 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2509.25164

[148] Roboflow, *Benchmarks*, Accessed: 2026-03-07, 2025. [Online]. Available: https://rfdetr.roboflow.com/1.5.1/learn/benchmarks/

[149] T.-D. Le et al., *Onboard satellite image classification for earth observation: A comparative study of vit models*, 2025. arXiv: 2409.03901 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2409.03901

[150] G. A. Pereira and M. Hussain, *A review of transformer-based models for computer vision tasks: Capturing global context and spatial relationships*, 2024. arXiv: 2408.15178 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2408.15178

[151] M. Caron et al., *Emerging properties in self-supervised vision transformers*, 2021. arXiv: 2104.14294 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2104.14294

[152] L. Wang and K.-J. Yoon, "Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 6, pp. 3048–3068, Jun. 2022, ISSN: 1939-3539. DOI: 10.1109/tpami.2021.3055564

[153] Z. Chen, C. Tang, and L. Xiong, *Hgnet: A hierarchical feature guided network for occupancy flow field prediction*, 2024. arXiv: 2407.01097 [eess.SY]. [Online]. Available: https://arxiv.org/abs/2407.01097

[154] N. Shazeer, *Glu variants improve transformer*, 2020. arXiv: 2002.05202 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2002.05202

[155] B. Zhang and R. Sennrich, *Root mean square layer normalization*, 2019. arXiv: 1910.07467 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1910.07467

[156] S. Huang, Z. Lu, X. Cun, Y. Yu, X. Zhou, and X. Shen, *Deim: Detr with improved matching for fast convergence*, 2025. arXiv: 2412.04234 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2412.04234

[157] Y. Peng, H. Li, P. Wu, Y. Zhang, X. Sun, and F. Wu, *D-fine: Redefine regression task in detrs as fine-grained distribution refinement*, 2024. arXiv: 2410.13842 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2410.13842

[158] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, *Generalized intersection over union: A metric and a loss for bounding box regression*, 2019. arXiv: 1902.09630 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1902.09630

[159] I. Shenfeld, M. Damani, J. Hübotter, and P. Agrawal, *Self-distillation enables continual learning*, 2026. arXiv: 2601.19897 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2601.19897

[160] Y. Li, R. Cheng, C. Zhang, M. Chen, H. Liang, and Z. Wang, "Dynamic mosaic algorithm for data augmentation," *Mathematical Biosciences and Engineering*, vol. 20, no. 4, pp. 7193–7216, 2023, ISSN: 1551-0018. DOI: 10.3934/mbe.2023311 [Online]. Available: https://www.aimspress.com/article/doi/10.3934/mbe.2023311

[161] PyTorch Contributors, *Transforming images, videos, boxes and more*, https://docs.pytorch.org/vision/main/transforms.html, Torchvision main documentation, version main (0.25.0a0+1e53952). Accessed: 2026-03-22, 2026.

[162] G. Mongaras, "Yolox explanation — mosaic and mixup for data augmentation," *Medium*, 2022, Accessed: 2023-03-08. [Online]. Available: https://gmongaras.medium.com/yolox-explanation-mosaic-and-mixup-for-data-augmentation-3839465a3adf

[163] Y. Shao et al., "Enhancing adversarial attacks with resize-invariant and logical ensemble," *Neural Networks*, vol. 173, p. 106 194, 2024, ISSN: 0893-6080. DOI: 10.1016/j.neunet.2024.106194 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608024001187

[164] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," *CoRR*, vol. abs/1710.09412, 2017. arXiv: 1710.09412. [Online]. Available: http://arxiv.org/abs/1710.09412

[165] P. Shyam, S. S. Sengar, K. Yoon, and K. Kim, "Evaluating COPY-BLEND augmentation for low level vision tasks," *CoRR*, vol. abs/2103.05889, 2021. arXiv: 2103.05889. [Online]. Available: https://arxiv.org/abs/2103.05889

[166] Y. Ma and F. Rusu, *Heterogeneous cpu+gpu stochastic gradient descent algorithms*, 2020. arXiv: 2004.08771 [cs.DC]. [Online]. Available: https://arxiv.org/abs/2004.08771

[167] LatLong.net. "Latlong.net — latitude and longitude finder." Accessed: 2026-02-28. [Online]. Available: https://www.latlong.net

[168] Time and Date AS. "Sunrise and sunset for rome, wallonia, belgium — june 2026." Accessed: 2026-02-28. [Online]. Available: https://www.timeanddate.com/sun/@2787825?month=6

[169] J. Ledo et al., "3d electrical resistivity of gran canaria island using magnetotelluric data," *Geothermics*, vol. 89, p. 101 945, 2021, ISSN: 0375-6505. DOI: 10.1016/j.geothermics.2020.101945 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0375650520302376

[170] R. Agramanisti Azdy and F. Darnis, "Use of haversine formula in finding distance between temporary shelter and waste end processing sites," *Journal of Physics: Conference Series*, vol. 1500, no. 1, p. 012 104, Apr. 2020. DOI: 10.1088/1742-6596/1500/1/012104

[171] W. Emery and A. Camps, "Chapter 5 - radar," in *Introduction to Satellite Remote Sensing*, W. Emery and A. Camps, Eds., Elsevier, 2017, pp. 291–453, ISBN: 978-0-12-809254-5. DOI: 10.1016/B978-0-12-809254-5.00005-1 [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128092545000051

[172] Labriji, H., Herscovici-Schiller, O., and Cassaing, F., "Computation of the lateral shift due to atmospheric refraction," *A&A*, vol. 662, A61, 2022. DOI: 10.1051/0004-6361/202142338

[173] European Space Agency, *Snap toolbox online help: Import worldview-2 products*, Accessed: 9 March 2026, ESA Sentinel Application Platform (SNAP), 2024. [Online]. Available: https://step.esa.int/main/wp-content/help/versions/11.0.0/snap-toolboxes/eu.esa.opt.opttbx.worldview2.reader/worldview2/worldview2.html

[174] R. Eke, T. Betts, and G. R., "Spectral irradiance effects on the outdoor performance of photovoltaic modules," *Renewable and Sustainable Energy Reviews*, vol. 69, pp. 429–434, 2017, ISSN: 1364-0321. DOI: 10.1016/j.rser.2016.10.062 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032116307079

[175] A. T. Mecherikunnel and J. C. Richmond, "Spectral distribution of solar radiation," National Aeronautics and Space Administration, Goddard Space Flight Center, Greenbelt, Maryland, NASA Technical Memorandum NASA TM-82021, Sep. 1980. [Online]. Available: https://ntrs.nasa.gov/api/citations/19810016493/downloads/19810016493.pdf

[176] R. Zhou et al., "Near-infrared photoactive semiconductor quantum dots for solar cells," *Advanced Energy Materials*, vol. 11, Sep. 2021. DOI: 10.1002/aenm.202101923

[177] H. Deneke, A. Feijt, and R. Roebeling, "Estimating surface solar irradiance from meteosat seviri-derived cloud properties," *Remote Sensing of Environment*, vol. 112, no. 6, pp. 3131–3141, 2008, ISSN: 0034-4257. DOI: 10.1016/j.rse.2008.03.012 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425708001107

[178] P.-W. Zhai et al., "Water-leaving contribution to polarized radiation field over ocean," *Optics Express*, vol. 25, no. 16, A689–A708, Aug. 2017. DOI: 10.1364/OE.25.00A689

[179] W. Su, T. P. Charlock, and K. Rutledge, "Observations of reflectance distribution around sunglint from a coastal ocean platform," *Appl. Opt.*, vol. 41, no. 35, pp. 7369–7383, Dec. 2002. DOI: 10.1364/AO.41.007369 [Online]. Available: https://opg.optica.org/ao/abstract.cfm?URI=ao-41-35-7369

[180] N. Longépé et al., "ΦSat-2 mission overview for the orbitalai challenge," European Space Agency (ESA), Technical Note, version 1.0, Feb. 15, 2023, ESA UNCLASSIFIED – Releasable to the Public. Accessed: Mar. 10, 2026. [Online]. Available: https://challenges.philab.esa.int/wp-content/uploads/2023/02/Phisat-2_Mission_Overview_Web.pdf

[181] N. Melega et al., "Development and implementation of the Φ-sat-2 mission," in *Small Satellites Systems and Services Symposium (4S 2024)*, M. Petrozzi-Ilstad, Ed., International Society for Optics and Photonics, vol. 13546, SPIE, 2025, p. 135462X. DOI: 10.1117/12.3062624

[182] Insula. "Insula perception Φ-sat-2 database." Redirects to the Φsat-2 data access page, Accessed: Mar. 10, 2026. [Online]. Available: https://phisat2.insula.earth/

[183] IAU Centre for the Protection of Dark and Quiet Sky from Satellite Constellation Interference. "Tle data access." SatChecker documentation, stable version, Accessed: Mar. 10, 2026. [Online]. Available: https://satchecker.readthedocs.io/en/stable/tools%5C_tle.html

[184] M. M. Abbas, A. Melesse, L. Scinto, and J. Rehage, "Satellite estimation of chlorophyll-a using moderate resolution imaging spectroradiometer (modis) sensor in shallow coastal water bodies: Validation and improvement," *Water*, vol. 11, Aug. 2019. DOI: 10.3390/w11081621

[185] A. Bannari, T. S. Ali, and A. Abahussain, "The capabilities of sentinel-msi (2a/2b) and landsat-oli (8/9) in seagrass and algae species differentiation using spectral reflectance," *Ocean Science*, vol. 18, no. 2, pp. 361–388, 2022. DOI: 10.5194/os-18-361-2022 [Online]. Available: https://os.copernicus.org/articles/18/361/2022/

[186] M. J. McCarthy et al., "Automated remote sensing tools to counter illicit maritime activity: Vessel detection, bathymetry and topography from worldview imagery," *International Journal of Remote Sensing*, vol. 44, no. 19, pp. 6117–6130, 2023. DOI: 10.1080/01431161.2023.2247525

[187] ICC International Maritime Bureau, "2023 annual imb piracy and armed robbery report," International Maritime Bureau (IMB), London, United Kingdom, Tech. Rep., Jan. 2024, Report No. 1253, published by the Japan P&I Club on behalf of the IMB. [Online]. Available: https://www.piclub.or.jp/wp-content/uploads/2024/01/No.1253_2023-Annual-IMB-Piracy-and-Armed-Robbery-Report.pdf

[188] G. Pramod, K. Nakamura, T. J. Pitcher, and L. Delagran, "Estimates of illegal and unreported fish in seafood imports to the usa," *Marine Policy*, vol. 48, pp. 102–113, 2014, ISSN: 0308-597X. DOI: 10.1016/j.marpol.2014.03.019

[189] C. Vincent-Lambert, A. Pretorius, and B. Van Tonder, "Use of unmanned aerial vehicles in wilderness search and rescue operations: A scoping review," *Wilderness and Environmental Medicine*, vol. 34, no. 4, pp. 580–588, 2023, ISSN: 1080-6032. DOI: 10.1016/j.wem.2023.08.022

[190] M. Kalamandeen et al., "Pervasive rise of small-scale deforestation in amazonia," *Scientific Reports*, vol. 8, no. 1, Jan. 2018, ISSN: 2045-2322. DOI: 10.1038/s41598-018-19358-2

[191] L. Blanc, V. Gond, and D. Ho Tong Minh, "2 - remote sensing and measuring deforestation," in *Land Surface Remote Sensing*, N. Baghdadi and M. Zribi, Eds., Elsevier, 2016, pp. 27–53, ISBN: 978-1-78548-105-5. DOI: 10.1016/B978-1-78548-105-5.50002-5

[192] Y. Gao, M. Skutsch, J. Paneque-Gálvez, and A. Ghilardi, "Remote sensing of forest degradation: A review," *Environmental Research Letters*, vol. 15, no. 10, p. 103 001, Sep. 2020, ISSN: 1748-9326. DOI: 10.1088/1748-9326/abaad7

[193] C. Finn, F. Grattarola, and D. Pincheira-Donoso, "More losers than winners: Investigating anthropocene defaunation through the diversity of population trends," *Biological Reviews*, vol. 98, no. 5, pp. 1732–1748, 2023. DOI: 10.1111/brv.12974

[194] S. Gupta et al., "Wildlife conservation and management: Challenges and strategies," *UTTAR PRADESH JOURNAL OF ZOOLOGY*, vol. 44, pp. 280–286, Dec. 2023. DOI: 10.56557/UPJOZ/2023/v44i243840

[195] M. Lee, Y. Kim, D. Nam, and K. Cho, "Impacts of the accumulated extinction of endangered species on stream food webs," *Global Ecology and Conservation*, vol. 48, e02747, 2023, ISSN: 2351-9894. DOI: 10.1016/j.gecco.2023.e02747

[196] M. C. Haller, D. Honegger, and P. A. Catalan, "Rip current observations via marine radar," *Journal of Waterway, Port, Coastal, and Ocean Engineering*, vol. 140, no. 2, pp. 115–124, 2014. DOI: 10.1061/(ASCE)WW.1943-5460.0000229

[197] N. Rampal, T. Shand, A. Wooler, and C. Rautenbach, "Interpretable deep learning applied to rip current detection and localization," *Remote Sensing*, vol. 14, no. 23, 2022, ISSN: 2072-4292. DOI: 10.3390/rs14236048

[198] M. Moulton, G. Dusek, S. Elgar, and B. Raubenheimer, "Comparison of rip current hazard likelihood forecasts with observed rip current speeds," *Weather and Forecasting*, vol. 32, no. 4, pp. 1659–1666, 2017. DOI: 10.1175/WAF-D-17-0076.1

[199] A. Sun and K. Yang, *Rip current detection in nearshore areas through uav video analysis with almost local-isometric embedding techniques on sphere*, 2024. arXiv: 2304.11783 [eess.IV]. [Online]. Available: https://arxiv.org/abs/2304.11783

[200] A. de Silva, I. Mori, G. Dusek, J. Davis, and A. Pang, "Automated rip current detection with region based convolutional neural networks," *Coastal Engineering*, vol. 166, p. 103 859, 2021, ISSN: 0378-3839. DOI: 10.1016/j.coastaleng.2021.103859

[201] X. Zhao and K. Jia, "Cloud removal in remote sensing using sequential-based diffusion models," *Remote Sensing*, vol. 15, no. 11, 2023, ISSN: 2072-4292. DOI: 10.3390/rs15112861

[202] B. E. Gryspeerdt, *Where is the cloudiest place on Earth? (Part 2 - Satellites)*, Mar. 2021. Accessed: Mar. 14, 2025. [Online]. Available: https://www.cloudsandclimate.com/blog/where_is_cloudiest_part2/

[203] M. Quiñones and M. Vissers, "K&c science report – phase 1: Tropical forest and wetlands mapping, case study borneo," Wageningen University, Dept. of Environmental Sciences, Wageningen, The Netherlands, Tech. Rep., 2023. [Online]. Available: https://www.researchgate.net/publication/255616914_KC_Science_Report_-_Phase_1_Tropical_Forest_and_Wetlands_Mapping_Case_Study_Borneo

[204] A. Meraner, P. Ebel, X. X. Zhu, and M. Schmitt, "Cloud removal in sentinel-2 imagery using a deep residual neural network and sar-optical data fusion," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 166, pp. 333–346, 2020, ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2020.05.013 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271620301398

[205] European Space Agency. "Trio of Sentinel satellites map methane super-emitters," Accessed: Nov. 11, 2024. [Online]. Available: https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Trio_of_Sentinel_satellites_map_methane_super-emitters

[206] E. Dowd et al., "First validation of high-resolution satellite-derived methane emissions from an active gas leak in the uk," *Atmospheric Measurement Techniques*, vol. 17, no. 5, pp. 1599–1615, 2024. DOI: 10.5194/amt-17-1599-2024

[207] D. Etkin, "Analysis of oil spill trends in the united states and worldwide," *International Oil Spill Conference Proceedings*, vol. 2001, Mar. 2001. DOI: 10.7901/2169-3358-2001-2-1291

[208] M. Fingas and C. E. Brown, "Chapter 6 - review of oil spill remote sensing: The current state of the art," in *Oil Spill Science and Technology (Third Edition)*, M. Fingas, Ed., Third Edition, Elsevier, 2025, pp. 309–358, ISBN: 978-0-443-21703-6. DOI: 10.1016/B978-0-443-21703-6.00003-5

[209] Q. Jiang, M. Ji, J. Wang, and P. Sun, "Remote sensing methods for striped marine oil spill detection in narrow ship channels," *Ocean Engineering*, vol. 289, p. 116 162, 2023, ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2023.116162

[210] M. W. Jones et al., "State of wildfires 2023–2024," *Earth System Science Data*, vol. 16, no. 8, pp. 3601–3685, 2024. DOI: 10.5194/essd-16-3601-2024

[211] T. Stelloh, M. Lenthang, R. Cohen, and P. Helsel. "California wildfires: What we know about the palisades and eaton fires in los angeles," Accessed: Jan. 22, 2025. [Online]. Available: https://www.nbcnews.com/news/us-news/california-wildfires-what-we-know-palisades-eaton-los-angeles-rcna188239

[212] Y. Chen, D. C. Morton, and J. T. Randerson, "Remote sensing for wildfire monitoring: Insights into burned area, emissions, and fire dynamics," *One Earth*, vol. 7, no. 6, pp. 1022–1028, 2024, ISSN: 2590-3322. DOI: 10.1016/j.oneear.2024.05.014

[213] G. H. de Almeida Pereira, A. M. Fusioka, B. T. Nassu, and R. Minetto, "Active fire detection in landsat-8 imagery: A large-scale dataset and a deep-learning study," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 178, pp. 171–186, 2021, ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2021.06.002 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092427162100160X

[214] K. A. Duda and B. K. Jones, "Usgs remote sensing coordination for the 2010 haiti earthquake," *Photogrammetric Engineering and Remote Sensing*, vol. 77, no. 9, pp. 899–907, Sep. 1, 2011, ISSN: 0099-1112. DOI: 10.14358/PERS.77.9.899

[215] A. Karaer et al., "Remote sensing-based comparative damage assessment of historical storms and hurricanes in northwestern florida," *International Journal of Disaster Risk Reduction*, vol. 72, p. 102 857, 2022, ISSN: 2212-4209. DOI: 10.1016/j.ijdrr.2022.102857

[216] Inter-Agency Space Debris Coordination Committee, "IADC report on the status of the space debris environment," Tech. Rep. AC.105/C.1/2024/CRP.16, Jan. 2024. [Online]. Available: https://www.unoosa.org/res/oosadoc/data/documents/2024/aac_105c_12024crp/aac_105c_12024crp_16_0_html/AC105_C1_2024_CRP16E.pdf

[217] D. Kessler, N. Johnson, J.-C. Liou, and M. Matney, "The kessler syndrome: Implications to future space operations," *Advances in the Astronautical Sciences*, vol. 137, Jan. 2010.

[218] P. Jharbade and M. Dixit, "Detecting space debris using deep learning algorithms: A survey," in *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2022, pp. 883–890. DOI: 10.1109/ICIRCA54612.2022.9985622

[219] D. Mehrholz et al., "Detecting, tracking and imaging space debris," Tech. Rep., Feb. 2002. [Online]. Available: https://www.esa.int/esapub/bulletin/bullet109/chapter16_bul109.pdf

[220] Z. Zhang, C. Deng, and Z. Deng, "A diverse space target dataset with multidebris and realistic on-orbit environment," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 9102–9114, 2022. DOI: 10.1109/JSTARS.2022.3203042

[221] N. Alkharji, H. Almazrouei, S. Alzaabi, A. B. Nassif, M. Elsalhy, and M. A. Talib, "Deep learning algorithms in aircraft detection and classification: An analytical survey," in *2024 17th International Conference on Development in eSystem Engineering (DeSE)*, 2024, pp. 568–573. DOI: 10.1109/DeSE63988.2024.10912016

[222] S. Merugu and K. Jain, "Change detection and estimation of illegal mining using satellite images," in *Proceedings of 2nd International Conference on Innovations in Electronics and Communication Engineering (ICIECE-2013)*, Guru Nanak Institutions, Hyderabad, Hyderabad, India, Aug. 2013, p. 246. [Online]. Available: https://www.researchgate.net/publication/260790890_Change_Detection_and_Estimation_of_Illegal_Mining_using_Satellite_Images

[223] R. N. Torres and P. Fraternali, "Aerialwaste dataset for landfill discovery in aerial and satellite images," *Scientific Data*, vol. 10, no. 1, p. 63, 2023, ISSN: 2052-4463. DOI: 10.1038/s41597-023-01976-9

[224] S. Syafrudin et al., "Analysis of factors influencing illegal waste dumping generation using gis spatial regression methods," *Sustainability*, vol. 15, p. 1926, Jan. 2023. DOI: 10.3390/su15031926

[225] V. Klemas, "Remote sensing of algal blooms: An overview with case studies," *Journal of Coastal Research*, vol. 28, pp. 34–43, Jan. 2012. DOI: 10.2112/JCOASTRES-D-11-00051.1

[226] C. Hou, M. L. Chu, and J. A. Guzman, "Risk assessment of harmful algal blooms (hab) occurrence in the agroecosystem: A hydro-ecologic modeling framework and environmental risk matrix," *Ecological Indicators*, vol. 145, p. 109 617, 2022, ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2022.109617

[227] J. Park, S. Khanal, K. Zhao, and K. Byun, "Remote sensing of chlorophyll-a and water quality over inland lakes: How to alleviate geo-location error and temporal discrepancy in model training," *Remote Sensing*, vol. 16, no. 15, 2024, ISSN: 2072-4292. DOI: 10.3390/rs16152761

[228] S. Rolim, B. K Veettil, A. Vieiro, A. Kessler, and C. Gonzatti, "Remote sensing for mapping algal blooms in freshwater lakes: A review," *Environmental Science and Pollution Research*, vol. 30, Jan. 2023. DOI: 10.1007/s11356-023-25230-2

[229] United States Environmental Protection Agency, *Sargassum inundation events (sies): Impacts on human health*, 2025. Accessed: Jan. 13, 2025. [Online]. Available: https://www.epa.gov/habs/sargassum-inundation-events-sies-impacts-human-health

[230] R. C. Gatti et al., "The number of tree species on earth," *Proceedings of the National Academy of Sciences*, vol. 119, no. 6, e2115329119, 2022. DOI: 10.1073/pnas.2115329119

[231] U. Andriolo and G. Gonçalves, "How much does marine litter weigh? a literature review to improve monitoring, support modelling and optimize clean-up activities," *Environmental Pollution*, vol. 361, p. 124 863, 2024, ISSN: 0269-7491. DOI: 10.1016/j.envpol.2024.124863

[232] R. J. van der A, J. Ding, and H. Eskes, "Monitoring european anthropogenic no_x emissions from space," *Atmospheric Chemistry and Physics*, vol. 24, no. 13, pp. 7523–7534, 2024. DOI: 10.5194/acp-24-7523-2024

[233] O. Bolivar, "Gdp nowcasting: A machine learning and remote sensing data-based approach for bolivia," *Latin American Journal of Central Banking*, vol. 5, no. 3, p. 100 126, 2024, ISSN: 2666-1438. DOI: 10.1016/j.latcb.2024.100126

[234] N. Caradot, P. R. Sampaio, A. S. Guilbert, H. Sonnenberg, V. Parez, and V. Dimova, "Using deterioration modelling to simulate sewer rehabilitation strategy with low data availability," *Water Science and Technology*, vol. 83, no. 3, pp. 631–640, Dec. 2020, ISSN: 0273-1223. DOI: 10.2166/wst.2020.604 eprint: https://iwaponline.com/wst/article-pdf/83/3/631/854192/wst083030631.pdf.

[235] A. O. Sojobi and T. Zayed, "Impact of sewer overflow on public health: A comprehensive scientometric analysis and systematic review," *Environmental Research*, vol. 203, p. 111 609, 2022, ISSN: 0013-9351. DOI: 10.1016/j.envres.2021.111609 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0013935121009038

[236] I. S. Wing, E. De Cian, and M. N. Mistry, "Global vulnerability of crop yields to climate change," *Journal of Environmental Economics and Management*, vol. 109, p. 102 462, 2021, ISSN: 0095-0696. DOI: 10.1016/j.jeem.2021.102462

[237] P. Kumar, P. Tiwari, A. Biswas, and P. Kumar Srivastava, "Spatio-temporal assessment of soil salinization utilizing remote sensing derivatives, and prediction modeling: Implications for sustainable development," *Geoscience Frontiers*, vol. 15, no. 6, p. 101 881, 2024, ISSN: 1674-9871. DOI: 10.1016/j.gsf.2024.101881

[238] S. Surujdeo-Maharaj, T. N. Sreenivasan, L. A. Motilal, and P. Umaharan, "Black pod and other phytophthora induced diseases of cacao: History, biology, and control," in *Cacao Diseases: A History of Old Enemies and New Encounters*, B. A. Bailey and L. W. Meinhardt, Eds. Cham: Springer International Publishing, 2016, pp. 213–266, ISBN: 978-3-319-24789-2. DOI: 10.1007/978-3-319-24789-2_7

[239] A. A. Appiah, "Evaluation of progress in cocoa crop protection and management," in *Shifting Frontiers of Theobroma Cacao*, S. O. Agele and O. S. Ibiremo, Eds., Rijeka: IntechOpen, 2023, ch. 7. DOI: 10.5772/intechopen.112642

[240] F. O. Anyimah, E. M. Osei Jnr, and C. Nyamekye, "Detection of stress areas in cocoa farms using gis and remote sensing: A case study of offinso municipal & offinso north district, ghana," *Environmental Challenges*, vol. 4, p. 100 087, 2021, ISSN: 2667-0100. DOI: 10.1016/j.envc.2021.100087

[241] C. Eijgenraam et al., "Economically efficient standards to protect the netherlands against flooding," *Interfaces*, vol. 44, pp. 7–21, Feb. 2014. DOI: 10.1287/inte.2013.0721

[242] A. Shastry, E. Carter, B. Coltin, R. Sleeter, S. McMichael, and J. Eggleston, "Mapping floods from remote sensing data and quantifying the effects of surface obstruction by clouds and vegetation," *Remote Sensing of Environment*, vol. 291, p. 113 556, 2023, ISSN: 0034-4257. DOI: 10.1016/j.rse.2023.113556

[243] N. Denissova, S. Nurakynov, O. Petrova, D. Chepashev, G. Daumova, and A. Yelisseyeva, "Remote sensing techniques for assessing snow avalanche formation factors and building hazard monitoring systems," *Atmosphere*, vol. 15, no. 11, 2024, ISSN: 2073-4433. DOI: 10.3390/atmos15111343

[244] J. P. Rafferty, "Hurricane helene," Jan. 2025. Accessed: Jan. 13, 2025. [Online]. Available: https://www.britannica.com/event/Hurricane-Helene

[245] R. Cooper, *Hurricane helene - dna*, 2025. Accessed: Jan. 13, 2025. [Online]. Available: https://www.osbm.nc.gov/hurricane-helene-dna/open#:~:text=Helene%20is%20one%20of%20the,confirmed%20deaths%20in%20North%20Carolina.&text=Large%2Dscale%20damage%20to%20homes,experienced%20severe%20or%20catastrophic%20impacts

[246] L. Ricciardulli et al., "Remote sensing and analysis of tropical cyclones: Current and emerging satellite sensors," *Tropical Cyclone Research and Review*, vol. 12, no. 4, pp. 267–293, 2023, ISSN: 2225-6032. DOI: 10.1016/j.tcrr.2023.12.003

[247] E. Larsen, "Predicting solar flares with remote sensing and machine learning," Oct. 2021. DOI: 10.48550/arXiv.2110.07658

[248] Rochus, P. et al., "The solar orbiter eui instrument: The extreme ultraviolet imager," *Astronomy and Astrophysics*, vol. 642, A8, 2020. DOI: 10.1051/0004-6361/201936663

[249] S. Narayana, R. V. Prasad, and T. V. Prabhakar, "Sos: Isolated health monitoring system to save our satellites," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21, Virtual Event, Wisconsin: Association for Computing Machinery, 2021, pp. 283–295, ISBN: 9781450384438. DOI: 10.1145/3458864.3466862

[250] J. P. Davis, J. P. Mayberry, J. P. Penn, C. F. S. POLICY, and STRATEGY, "On-Orbit Servicing: Inspection, repair, refuel, upgrade, and assembly of satellites in space," Tech. Rep., Apr. 2019. [Online]. Available: https://aerospace.org/sites/default/files/2019-05/Davis-Mayberry-Penn_OOS_04242019.pdf

[251] Z. Hou, B. Jiao, and Z. Dang, "Relative orbit design of cubesats for on-orbit visual inspection of china space station," *Advances in Space Research*, vol. 73, no. 1, pp. 170–186, 2024, ISSN: 0273-1177. DOI: 10.1016/j.asr.2023.10.010

[252] A. Scorsoglio, A. D'Ambrosio, L. Ghilardi, and F. Curti, "Safe lunar landing via images: A reinforcement meta-learning application to autonomous hazard avoidance and landing," in *2020 AAS/AIAA Astrodynamics Specialist Conference*, Jul. 2020. [Online]. Available: https://www.researchgate.net/publication/343650361_Safe_lunar_landing_via_images_a_reinforcement_meta-learning_application_to_autonomous_hazard_avoidance_and_landing

[253] R. Brockers et al., "Autonomous safe landing site detection for a future mars science helicopter," in *2021 IEEE Aerospace Conference (50100)*, 2021, pp. 1–8. DOI: 10.1109/AERO50100.2021.9438289

[254] C. L. Marcus, R. Zanetti, and T. P. Setterfield, "Landing site mapping and selection with quadtree map using planar elements," *Journal of Guidance, Control, and Dynamics*, vol. 47, no. 7, pp. 1283–1297, 2024. DOI: 10.2514/1.G007801

[255] Q. Meng et al., "High resolution imaging camera (hiric) on china's first mars exploration tianwen-1 mission," *Space Science Reviews*, vol. 217, Apr. 2021. DOI: 10.1007/s11214-021-00823-w

[256] S. M. Mousavi Kahaki and M. J. Nordin, "Highway traffic incident detection using high-resolution aerial remote sensing imagery," *Journal of Computer Science*, vol. 7, pp. 949–953, May 2011. DOI: 10.3844/jcssp.2011.949.953

[257] M. Langbroek, *Mh17: Mogelijkheid van militaire space based (satelliet) observaties van het afvuren van een grond-luchtraket*, Memo presented at the roundtable discussion 'Radar en satellietgegevens', Vaste Kamercommissie BuZa, Jan. 2016.

[258] A. H. Altieri and R. J. Diaz, "Chapter 24 - dead zones: Oxygen depletion in coastal ecosystems," in *World Seas: An Environmental Evaluation (Second Edition)*, C. Sheppard, Ed., Second Edition, Academic Press, 2019, pp. 453–473, ISBN: 978-0-12-805052-1. DOI: 10.1016/B978-0-12-805052-1.00021-8 [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128050521000218

[259] J. W. Morse and P. M. Eldridge, "A non-steady state diagenetic model for changes in sediment biogeochemistry in response to seasonally hypoxic/anoxic conditions in the "dead zone" of the louisiana shelf," *Marine Chemistry*, vol. 106, no. 1, pp. 239–255, 2007, Special issue: Dedicated to the memory of Professor Roland Wollast, ISSN: 0304-4203. DOI: 10.1016/j.marchem.2006.02.003 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304420306000375

[260] M. Dirscherl, A. J. Dietz, S. Dech, and C. Kuenzer, "Remote sensing of ice motion in antarctica - a review," *Remote Sensing of Environment*, vol. 237, p. 111 595, 2020, ISSN: 0034-4257. DOI: 10.1016/j.rse.2019.111595

[261] Y. Koo, H. Xie, H. Mahmoud, J. M. Iqrah, and S. F. Ackley, "Automated detection and tracking of medium-large icebergs from sentinel-1 imagery using google earth engine," *Remote Sensing of Environment*, vol. 296, p. 113 731, 2023, ISSN: 0034-4257. DOI: 10.1016/j.rse.2023.113731

[262] B. Evans, A. Faul, A. Fleming, D. G. Vaughan, and J. S. Hosking, "Unsupervised machine learning detection of iceberg populations within sea ice from dual-polarisation sar imagery," *Remote Sensing of Environment*, vol. 297, p. 113 780, 2023, ISSN: 0034-4257. DOI: 10.1016/j.rse.2023.113780

[263] S. W. Kerber, N. A. Duncan, G. F. L'Her, M. Bazilian, C. Elvidge, and M. R. Deinert, "Tracking electricity losses and their perceived causes using nighttime light and social media," *iScience*, vol. 26, no. 12, p. 108 381, 2023, ISSN: 2589-0042. DOI: 10.1016/j.isci.2023.108381

[264] Z. Shah, N. Klugman, G. Cadamuro, F.-C. Hsu, C. D. Elvidge, and J. Taneja, "The electricity scene from above: Exploring power grid inconsistencies using satellite data in accra, ghana," *Applied Energy*, vol. 319, p. 119 237, 2022, ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2022.119237

[265] A. Ganji, M. Zhang, and M. Hatzopoulou, "Traffic volume prediction using aerial imagery and sparse data from road counts," *Transportation Research Part C: Emerging Technologies*, vol. 141, p. 103 739, 2022, ISSN: 0968-090X. DOI: 10.1016/j.trc.2022.103739 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X22001735

[266] A. Grządziel, "Application of remote sensing techniques to identification of underwater airplane wreck in shallow water environment: Case study of the baltic sea, poland," *Remote Sensing*, vol. 14, no. 20, 2022, ISSN: 2072-4292. DOI: 10.3390/rs14205195

[267] K. J. Vrana et al., "Mapping rms titanic with gis: Implications for forensic investigations," *Marine Technology Society Journal*, vol. 46, no. 6, pp. 111–128, Nov. 1, 2012, ISSN: 0025-3324. DOI: doi:10.4031/MTSJ.46.6.8

[268] N. Bachagha, X. Wang, L. Luo, L. Li, H. Khatteli, and R. Lasaponara, "Remote sensing and gis techniques for reconstructing the military fort system on the roman boundary (tunisian section) and identifying archaeological sites," *Remote Sensing of Environment*, vol. 236, p. 111 418, 2020, ISSN: 0034-4257. DOI: 10.1016/j.rse.2019.111418

[269] I. G. Hendrawan et al., "Characteristics and distribution of stranded plastic pollution in bali conservation areas," *Marine Pollution Bulletin*, vol. 197, p. 115 770, 2023, ISSN: 0025-326X. DOI: 10.1016/j.marpolbul.2023.115770

[270] I. E. Napper et al., "Reaching new heights in plastic pollution—preliminary findings of microplastics on mount everest," *One Earth*, vol. 3, no. 5, pp. 621–630, 2020, ISSN: 2590-3322. DOI: 10.1016/j.oneear.2020.10.020

[271] Z. Xu, H. Zhang, Y. Wang, X. Wang, S. Xue, and W. Liu, "Dynamic detection of offshore wind turbines by spatial machine learning from spaceborne synthetic aperture radar imagery," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 5, pp. 1674–1686, 2022, ISSN: 1319-1578. DOI: 10.1016/j.jksuci.2022.02.027

[272] L. Gulbe, V. Caune, and G. Korats, "Urban area thermal monitoring: Liepaja case study using satellite and aerial thermal data," *International Journal of Applied Earth Observation and Geoinformation*, vol. 63, pp. 45–54, 2017, ISSN: 1569-8432. DOI: 10.1016/j.jag.2017.07.005

[273] Digital Matter. "Remote sensing livestock management," Accessed: Jan. 15, 2025. [Online]. Available: https://sense.digitalmatter.com/blog/remote-sensing-livestock-management

[274] Digital Matter. "Livestock monitoring in the alpes," Accessed: Jan. 13, 2025. [Online]. Available: https://www.digitalmatter.com/case-studies/livestock-monitoring-in-the-alpes/

[275] H. Alvan and H. Omar, "Overview of remote sensing techniques in earthquake prediction," *Journal of Engineering, Design and Technology*, vol. 9, pp. 164–177, Jul. 2011. DOI: 10.1108/17260531111151050

[276] S. Koshimura, L. Moya, E. Mas, and Y. Bai, "Tsunami damage detection with remote sensing: A review," *Geosciences*, vol. 10, no. 5, 2020, ISSN: 2076-3263. DOI: 10.3390/geosciences10050177

[277] A. Tronin, "Remote sensing and earthquakes: A review," *Physics and Chemistry of the Earth, Parts A/B/C*, vol. 31, no. 4, pp. 138–142, 2006, Recent Progress in Seismo Electromagnetics and Related Phenomena, ISSN: 1474-7065. DOI: 10.1016/j.pce.2006.02.024

[278] D. Nikolopoulos, D. Cantzos, A. Alam, S. Dimopoulos, and E. Petraki, "Electromagnetic and radon earthquake precursors," *Geosciences*, vol. 14, no. 10, 2024, ISSN: 2076-3263. DOI: 10.3390/geosciences14100271

[279] R. Stosius, G. Beyerle, A. Hoechner, J. Wickert, and J. Lauterjung, "The impact on tsunami detection from space using gnss-reflectometry when combining gps with glonass and galileo," *Advances in Space Research*, vol. 47, no. 5, pp. 843–853, 2011, Scientific applications of Galileo and other Global Navigation Satellite Systems - II, ISSN: 0273-1177. DOI: 10.1016/j.asr.2010.09.022

[280] S. Chaturvedi, P. Srivastava, and U. Guven, "A brief review on tsunami early warning detection using bpr approach and post analysis by sar satellite dataset," *Journal of Ocean Engineering and Science*, vol. 2, Apr. 2017. DOI: 10.1016/j.joes.2016.12.001

[281] C. H.K, K. C. Flynn, and A. J. Ashworth, "Advancements in remote sensing techniques for earthquake engineering: A review," *Earthquake Research Advances*, p. 100 352, 2024, ISSN: 2772-4670. DOI: 10.1016/j.eqrea.2024.100352

[282] N. Ahmad et al., "Earthquake prediction: A global review and local research," *Pakistan academy of Sciences 46(4):233-246.2009*, Jan. 2009.

[283] Z. Lu and P. Webley, "Special issue: Volcano remote sensing," *Remote Sensing*, vol. Special Issue, 2015, ISSN: 2072-4292. [Online]. Available: https://www.mdpi.com/journal/remotesensing/special_issues/volcano

[284] M. C. Garthwaite, V. L. Miller, S. Saunders, M. M. Parks, G. Hu, and A. L. Parker, "A simplified approach to operational insar monitoring of volcano deformation in low- and middle-income countries: Case study of rabaul caldera, papua new guinea," *Frontiers in Earth Science*, vol. 6, 2019, ISSN: 2296-6463. DOI: 10.3389/feart.2018.00240

[285] S.-W. Lee, S.-H. Yun, D. Kim, D. Lee, Y. Lee, and B. Schutz, "Real-time volcano monitoring using gnss single-frequency receivers," *Journal of Geophysical Research: Solid Earth*, vol. 120, n/a–n/a, Dec. 2015. DOI: 10.1002/2014JB011648

[286] D. Coppola et al., "Thermal remote sensing for global volcano monitoring: Experiences from the mirova system," *Frontiers in Earth Science*, vol. 7, 2020, ISSN: 2296-6463. DOI: 10.3389/feart.2019.00362

[287] C. Kern, S. Arellano, R. Campion, S. Hidalgo, and R. Kazahaya, "Editorial: Remote sensing of volcanic gas emissions from the ground, air, and space," *Frontiers in Earth Science*, vol. 11, 2023, ISSN: 2296-6463. DOI: 10.3389/feart.2023.1340395

[288] Y. Guo, N. Yin, W. Wu, J. Huang, and Q. Wang, "A remote sensing satellite lighting detection system with high detection rate and low complexity," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLIII-B3-2022, pp. 101–106, May 2022. DOI: 10.5194/isprs-archives-XLIII-B3-2022-101-2022

[289] S. Abarca, K. Corbosiero, and T. Galarneau, "An evaluation of the world wide lightning location network (wwlln) using the national lightning detection network (nldn) as ground truth," *Journal of Geophysical Research*, vol. 115, Sep. 2010. DOI: 10.1029/2009JD013411

[290] W. A. Lyons et al., "Different strokes: Researching the unusual lightning discharges associated with sprites and jets and atypical meteorological regimes," in *22nd International Lightning Detection Conference & 4th International Lightning Meteorology Conference*, Broomfield, Colorado, USA, Apr. 2012. [Online]. Available: https://www.researchgate.net/publication/301957344_Different_strokes_Researching_the_unusual_lightning_discharges_associated_with_sprites_and_jets_and_atypical_meteorological_regimes

[291] T. Ushio et al., "The global lightning and sprite measurement (glims) mission on international space station," *IEEJ Transactions on Fundamentals and Materials*, vol. 131, no. 12, pp. 971–976, 2011. DOI: 10.1541/ieejfms.131.971

[292] A. Bhaskar et al., "Auroramag: Twin explorer of asymmetry in aurora and solar wind-magnetosphere coupling," *Advances in Space Research*, 2024, ISSN: 0273-1177. DOI: 10.1016/j.asr.2024.05.067

[293] V. Kalb, B. Kosar, Y. Collado-Vega, and C. Davidson, "Aurora detection from nighttime lights for earth and space science applications," *Earth and Space Science*, vol. 10, no. 1, e2022EA002513, 2023. DOI: 10.1029/2022EA002513

[294] A. Vorobev, A. Soloviev, V. Pilipenko, and G. Vorobeva, "Interactive computer model for aurora forecast and analysis," *Solnechno-Zemnaya Fizika*, vol. 8, pp. 93–100, Jun. 2022. DOI: 10.12737/szf-82202213

[295] T. Nishiyama et al., "The first simultaneous spectroscopic and monochromatic imaging observations of short-wavelength infrared aurora of N_2^+ meinel (0, 0) band at 1.1 μm with incoherent scatter radar," *Earth, Planets and Space*, vol. 76, no. 1, Feb. 2024, ISSN: 1880-5981. DOI: 10.1186/s40623-024-01969-x

[296] E. J. Fusco, J. T. Finn, J. T. Abatzoglou, J. K. Balch, S. Dadashi, and B. A. Bradley, "Detection rates and biases of fire observations from modis and agency reports in the conterminous united states," *Remote Sensing of Environment*, vol. 220, pp. 30–40, 2019, ISSN: 0034-4257. DOI: 10.1016/j.rse.2018.10.028

[297] K. Coskuner, "Assessing the performance of modis and viirs active fire products in the monitoring of wildfires: A case study in turkey," *iForest - Biogeosciences and Forestry*, no. 2, pp. 85–94, 2022. DOI: 10.3832/ifor3754-015

[298] A. Dimitrakopoulos, C. Gogi, G. Stamatelos, and I. Mitsopoulos, "Statistical analysis of the fire environment of large forest fires (> 1000 ha) in greece," *Polish Journal of Environmental Studies*, vol. 20, pp. 327–332, Jan. 2011.

[299] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. Grammalidis, "A review on early forest fire detection systems using optical remote sensing," *Sensors*, vol. 20, no. 22, 2020, ISSN: 1424-8220. DOI: 10.3390/s20226442

[300] Muon Space and Earth Fire Alliance. "Muon space and earth fire alliance unveil firesat constellation: A revolutionary space mission to transform global wildfire response," Accessed: Jan. 22, 2024. [Online]. Available: https://www.muonspace.com/press/muon-space-and-earth-fire-alliance-unveil-firesat-constellation-a-revolutionary-space-mission-to-transform-global-wildfire-response

[301] H. Yao, Z. Yang, G. Zhang, and F. Liu, "Forest fire detection based on spatial characteristics of surface temperature," *Remote Sensing*, vol. 16, no. 16, 2024, ISSN: 2072-4292. DOI: 10.3390/rs16162945

[302] M. Affek and J. Szymański, "A survey on the datasets and algorithms for satellite data applications," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 16 078–16 099, 2024. DOI: 10.1109/JSTARS.2024.3424954

[303] A. Aaba, *Wildfire prediction dataset (satellite images)*, 2023. [Online]. Available: https://www.kaggle.com/datasets/abdelghaniaaba/wildfire-prediction-dataset

[304] G. Wang et al., "M4sfwd: A multi-faceted synthetic dataset for remote sensing forest wildfires detection," *Expert Systems with Applications*, vol. 248, p. 123 489, 2024, ISSN: 0957-4174. DOI: 10.1016/j.eswa.2024.123489

[305] N. Wright, J. M. Duncan, J. N. Callow, S. E. Thompson, and R. J. George, "Clouds2mask: A novel deep learning approach for improved cloud and cloud shadow masking in sentinel-2 imagery," *Remote Sensing of Environment*, vol. 306, p. 114 122, 2024, ISSN: 0034-4257. DOI: 10.1016/j.rse.2024.114122

[306] S. Mohajerani and P. Saeedi, "Cloud-Net: An End-To-End Cloud Detection Algorithm for Landsat 8 Imagery," in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 1029–1032. DOI: 10.1109/IGARSS.2019.8898776

[307] K. Lee, "Current status of satellite development and application," English, *Korean Journal of Remote Sensing*, vol. 40, no. 5, pp. 695–712, Oct. 2024, ISSN: 1225-6161, 2287-9307. DOI: 10.7780/kjrs.2024.40.5.2.2

[308] J. Boerkoel et al., "Advances in scheduling and planning for space missions," in *Proceedings of the International Workshop on Planning and Scheduling for Space (IWPSS)*, 2021. [Online]. Available: https://ai.jpl.nasa.gov/public/documents/papers/Boerkoel-IWPSS2021-paper-23.pdf

[309] A.-J. Gallego, A. Pertusa, and P. Gil, "Automatic ship classification from optical aerial images with convolutional neural networks," *Remote Sensing*, vol. 10, no. 4, 2018, ISSN: 2072-4292. DOI: 10.3390/rs10040511

[310] K. Rainey, S. Parameswaran, J. Harguess, and J. Stastny, "Vessel classification in overhead satellite imagery using learned dictionaries," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 8499, Oct. 2012. DOI: 10.1117/12.928875

[311] J. Chen, K. Chen, H. Chen, Z. Zou, and Z. Shi, "A degraded reconstruction enhancement-based method for tiny ship detection in remote sensing images with a new large-scale dataset," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022. DOI: 10.1109/TGRS.2022.3180894

[312] L. Bui et al., "Uow-vessel: A benchmark dataset of high-resolution optical satellite images for vessel detection and segmentation," in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024, pp. 4416–4424. DOI: 10.1109/WACV57701.2024.00437

[313] X. Pu, H. Jia, Y. Xin, F. Wang, and H. Wang, "Ship detection in low-quality sar images via an unsupervised domain adaption method," *Remote Sensing*, vol. 15, p. 3326, Jun. 2023. DOI: 10.3390/rs15133326

[314] O. Aleuy, S. Kutz, M. Mallory, and J. Provencher, "Wildlife health in environmental impact assessments: Are we missing a key metric?" *Environmental Reviews*, vol. 31, Nov. 2022. DOI: 10.1139/er-2022-0023

[315] I. Duporge, O. Isupova, S. Reece, D. W. Macdonald, and T. Wang, "Using very-high-resolution satellite imagery and deep learning to detect and count african elephants in heterogeneous landscapes," *Remote Sensing in Ecology and Conservation*, vol. 7, no. 3, pp. 369–381, 2021. DOI: 10.1002/rse2.195

[316] Z. Xu, T. Wang, A. K. Skidmore, and R. Lamprey, "A review of deep learning techniques for detecting animals in aerial and satellite images," *International Journal of Applied Earth Observation and Geoinformation*, vol. 128, p. 103 732, 2024, ISSN: 1569-8432. DOI: 10.1016/j.jag.2024.103732

[317] Y. Zhang and Z. Cai, "Ce-retinanet: A channel enhancement method for infrared wildlife detection in uav images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, 2023, ISSN: 0196-2892. DOI: 10.1109/TGRS.2023.3299651

[318] A. Winterl et al., "Remote sensing of emperor penguin abundance and breeding success," *Nature Communications*, vol. 15, no. 1, p. 4419, 2024. DOI: 10.1038/s41467-024-48239-8

[319] Y. Noboru, Y. Ozasa, and M. Tanaka, *Hyperspectral image dataset for individual penguin identification*, 2024. arXiv: 2405.14146 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2405.14146

[320] J. Wu, W. Xu, J. He, and M. Lan, "Yolo for penguin detection and counting based on remote sensing images," *Remote Sensing*, vol. 15, no. 10, 2023, ISSN: 2072-4292. DOI: 10.3390/rs15102598

[321] P. Fretwell, *Counting whales by satellite*, British Antarctic Survey, Mar. 2017. Accessed: Apr. 10, 2025. [Online]. Available: https://blog.maxar.com/earth-intelligence/2017/counting-whales-by-satellite

[322] C. A. Watt, C. G. Carlyle, C. A. Hornby, and B. A. H. Sherbo, "Eastern high arctic-baffin bay beluga whale (delphinapterus leucas) estuary abundance and use from space," *Frontiers in Mammal Science*, vol. 2, 2023, ISSN: 2813-4699. DOI: 10.3389/fmamm.2023.1208276

[323] E. N. Rodofili, L. Houegnigan, V. Lecours, G. Klarenberg, and E. Cuesta, "Automated detection of migrating gray whales and measurement of their bearing in satellite imagery," *Marine Mammal Science*, e13213, 2024. DOI: 10.1111/mms.13213

[324] M. Patel et al., "The influence of input image scale on deep learning-based beluga whale detection from aerial remote sensing imagery," Jul. 2023, pp. 5732–5734. DOI: 10.1109/IGARSS52108.2023.10281408

[325] B. Charry, E. Tissier, J. Iacozza, M. Marcoux, and C. A. Watt, "Mapping arctic cetaceans from space: A case study for beluga and narwhal," *PLOS ONE*, vol. 16, no. 8, pp. 1–14, Aug. 2021. DOI: 10.1371/journal.pone.0254380

[326] B. A. H. Sherbo, A. M. Belanger, B. Charry, and C. A. Watt, "Using satellite imagery to estimate abundance of cumberland sound beluga whales (delphinapterus leucas) in 2021," *Frontiers in Marine Science*, vol. 11, 2024, ISSN: 2296-7745. DOI: 10.3389/fmars.2024.1422907

[327] M. Thums, C. Jenner, K. Waples, C. Salgado Kent, and M. Meekan, "Humpback whale use of the kimberley; understanding and monitoring spatial distribution," Western Australian Marine Science Institution, Perth, Western Australia, Project Report Project 1.2.1, 2018, Report prepared for the Kimberley Marine Research Program, p. 78.

[328] V. Panigada, H. C. Cubaynes, P. T. Fretwell, and J. A. Jackson, *Annotated dataset of grey whales detected in very high resolution (vhr) satellite imagery of baja california sur lagoons in 2009, 2013 and 2015*, version 1.0, NERC EDS UK Polar Data Centre, 2024. DOI: 10.5285/2d8a4b45-e21f-47d4-8279-b4362489301e

[329] P. T. Fretwell, H. C. Cubaynes, and O. V. Shpak, "Satellite image survey of beluga whales in the southern kara sea," *Marine Mammal Science*, vol. 39, no. 4, pp. 1204–1214, 2023. DOI: 10.1111/mms.13044

[330] J. B. Stewart, J. M. Hudson, B. A. H. Sherbo, and C. A. Watt, "Estimating beluga whale abundance from space: Using drones to ground-validate vhr satellite imagery," *Remote Sensing in Ecology and Conservation*, vol. 10, no. 6, pp. 673–685, 2024. DOI: 10.1002/rse2.396

[331] A. A. Corrêa, J. H. Quoos, A. S. Barreto, K. R. Groch, and P. P. B. Eichler, "Use of satellite imagery to identify southern right whales (eubalaena australis) on a southwest atlantic ocean breeding ground," *Marine Mammal Science*, vol. 38, no. 1, pp. 87–101, 2022. DOI: 10.1111/mms.12847

[332] L. Houegnigan et al., "Wildlife and marine mammal spatial observatory: Observation and automated detection of southern right whales in multispectral satellite imagery," *bioRxiv*, 2022. DOI: 10.1101/2022.01.20.477141

[333] M. Hodul et al., "Individual north atlantic right whales identified from space," *Marine Mammal Science*, vol. 39, no. 1, pp. 220–231, 2023. DOI: 10.1111/mms.12971

A

Appendix A - Application Selection

This appendix documents the selection of the demonstration use case used throughout the thesis. Whale detection was selected as the representative AI-Based Tip and Cue application, and the material here provides the screening and trade-off analysis that led to that choice. The objective is to show that the final use is suitable for consistent evaluation within the defined benchmark framework.

A.1. Selection Criterion

The screening was performed using four criteria:

- **Tip and Cue Benefit:** The extent to which the application structurally benefits from large-area scanning combined with high-resolution confirmation, spatial uncertainty, and latency sensitivity.
- **Urgency:** The societal relevance, novelty, and whether existing systems already provide comparable operational capabilities.
- **Data Characteristics:** The suitability of the observable features with respect to off-nadir distortions, spectral dependency, and spatial pattern clarity.
- **Data Availability:** The availability of reproducible and accessible datasets at both high and moderate spatial resolution, including labelled data for AI-based evaluation.

Applications that did not satisfy the architectural requirements of Tip and Cue or could not be reliably evaluated within the defined simulation framework were excluded in an initial filtering step. Table A.1 summarizes all considered applications and their primary elimination categories.

The remaining candidates were subjected to a trade-off analysis based on the four evaluation criteria. The detailed scoring and sensitivity assessment are presented in the following sections.

Table A.1: Overview of the application elimination process, grouped by elimination category.

Elimination Category	Application	Constraint
Applications without Time Urgency	Illegal Land Activity Detection (Mining, Waste Disposal, Property Changes)	Long term changes
	Harmful Algal Bloom Detection	Multiple day time window
	Biodiversity Monitoring	Long term changes
	Marine Ecosystem Health Monitoring (Coral Reefs, Ocean Dead Zones)	Long term changes
	Global Oceanic Circulation Monitoring	Long term changes
	Plastic Soup Mapping	No urgent response needed
	Glacier Dynamics Monitoring	Long term changes
	Ground Movement and Geohazard Monitoring Soil, Infrastructure Instability, Sink Holes)	Long term changes
	Archaeological Site Detection	No urgent response needed
	Remote Trash Mapping	No urgent response needed
	Technology Inspection (Solar Panels, Windmills)	No urgent response needed
	Thermal Insulation Monitoring	No urgent response needed
Applications without Location Uncertainty	Accident Investigation	Human-tipped
	Power Grid Failure Detection	Human-tipped
	Traffic Flow Optimization	Known location
	Agriculture Monitoring (Soil and Crop Health, Disease Spread)	Human-cued
	Livestock Management	Human-tipped
	Flooding Dynamics Monitoring	Human-tipped
	Avalanche Prediction	Human-tipped
	Volcanic Eruption Prediction	Known location
	Damage Assessment	Human-tipped
Applications with More Efficient Tip and Cue Methods	Unauthorized Flight Detection	Aircraft radar more efficient
	Iceberg Detection	Marine radar more efficient
	Oil Spill Detection	Marine radar more efficient
	Solar Event Prediction	One satellite with two cameras is more efficient
	Landing Site Detection	One lander with two cameras is more efficient
Applications with Imaging Limitations	Deforestation Detection	Off-nadir angle omissions
	Search and Rescue Operations	Insufficient tipping tech
	Emission Profile Monitoring	High spectral dependency
	Methane Leak Detection	High spectral dependency
	Seismic Activity Prediction (Tsunamis, Earth Quakes)	Insufficient tipping tech
	Cyclone and Hurricane Prediction	Processing too complex
	Lightning Strike and TLE Mapping	Insufficient imaging tech
	Aurora Borealis Monitoring	High spectral dependency
Applications with Insufficient Data Availability	Military Operations	Restricted data access
	Rib Current Detection	No high swath / Tip satellite data
	Space Debris Monitoring	No simulation tools available
	Satellite Inspection	No data exists
Application Candidates	Wildfire Detection	
	Permanently Clouded Region Mapping	
	Illegal Maritime Activity Detection (Piracy, Illegal Fishing, Migration)	
	Wildlife Conservation	

After the elimination, four application candidates remained: Wildfire Detection, Permanently Clouded Region Mapping, Illegal Maritime Activity Detection, and Wildlife Monitoring. Then based on the four evaluation criteria defined: Tip and Cue Benefit, Urgency, Data Characteristics, and Data Availability, a trade-off analysis was performed.

The results are summarized in Table A.18. Each application was scored from 0 to 5 per main criterion, based on qualitative assessment of defined sub-criteria. The sub-criteria were evaluated using symbolic ratings: '+' (fully satisfied), 'o' (partially satisfied), and '-' (not satisfied). These symbolic ratings were converted into numerical contributions by mapping '+' to the maximum attainable points of the respective sub-criterion, 'o' to half of those points, and '-' to zero. Since each criterion ranges from 0 to 5 and all sub-criteria within one criterion contribute equally, the maximum contribution per sub-criterion equals 5 divided by the number of its sub-criteria. The individual contributions were summed and rounded to obtain the final criterion score.

The four main criteria were also weighted equally and summed to a total score. The highest-ranking

application was selected as demonstration use case for the AI-Based Tip and Cue simulation framework.

A.2. Application Scoring

Based on the evaluation criteria defined in section A.1, the applications Wildfire Detection, Permently Clouded Region Mapping, Illegal Martime Activities, and Wildlife Monitoring were scored, after which one final application was selected.

A.2.1. Wildfire Detection

Tip and Cue Benefit: 4

Wildfire detection aligns with several strengths of a Tip and Cue architecture. A large field of view is required to monitor fire outbreaks across remote and spatially extended regions. High spatial resolution is necessary for early localization and boundary delineation, particularly in the ignition phase.

Current satellite systems only partially meet these requirements. MODIS can detect fires as small as 50 m^2 within a 30 × 30 m pixel under favourable conditions, yet reported real-world detection rates range between 3.5–23.4% for wildfires larger than 1 ha [296]. VIIRS improves spatial accuracy at 375 m resolution, but case studies report detection rates between 1.3–25.6% of recorded ground-based fires [297].

Temporal responsiveness is critical. Large forest fires often require an initial attack within 30 minutes [298]. Current revisit and latency constraints — approximately 12 hours for VIIRS and 2–4 hours processing latency for MODIS — limit early intervention [299]. A Tip and Cue system could reduce response delay by combining wide-area monitoring with targeted high-resolution confirmation.

Multi-angle viewing, however, provides limited added value. At higher off-nadir angles, tree canopy obstruction may reduce visibility of early-stage fires. This constrains one of the architectural advantages of Tip and Cue.

Based on these considerations, wildfire detection receives a Tip and Cue benefit score of 4 (see Table A.2).

Table A.2: Sub-criteria scores for Tip and Cue benefit for Wildfire Detection.

Criterion	Justification	Score	Numerical Contribution
Large field of view	Yes, needed for detection in large, remote areas	+	0.83
High spatial resolution	Yes, required for early detection and fire localization	+	0.83
Low latency	Yes, critical for short initial attack windows	+	0.83
High temporal resolution	Yes, needed for frequent coverage of fire-prone zones	+	0.83
Long revisit time	Yes, needed to monitor fire spread for a long duration	+	0.83
Multi-angle view	No, does not give additional information and limits view due to tree obstruction	-	0
		Total	**4 (4.16)**

Urgency: 1

Despite the high societal impact of wildfires, the need for a new Tip and Cue system is limited. The planned FireSat constellation aims to deploy 50+ satellites with 80 m resolution, 20-minute revisit times, and the ability to detect hotspots as small as 5x5 m. Phase 1, launching in 2026, already fulfills many Tip and Cue goals [300]. On the model performance side, wildfire detection models are already highly developed. A review of 37 deep learning methods between 2018 and 2023 shows that most achieve over 90% accuracy in segmentation and detection tasks [29]. However, this performance is not reflected in real-world monitoring, given a 75% miss rate as previously discussed [297]. The performance gap can be caused due to a dataset imbalance, delayed detection of small fires, poor temporal resolution, and cloud coverage. While Tip and Cue can reduce latency and optimize the detection accross two satellites, its novelty is low, and the need is partially filled by upcoming dedicated systems. Table A.3 summarizes the score of 1 for the Urgency criterion.

Table A.3: Sub-criteria scores for urgency of AI-based Tip and Cue for Wildfire Detection.

Criterion	Justification	Score	Numerical Contribution
Tip and Cue system need	Maybe, existing technologies like FireSat constellation may make it redundant	o	0.83
Novelty	No, forest fire detection is well-researched with many models available	-	0
Currently achieved performances	High accuracy >90% already demonstrated in literature	-	0
		Total	**0.83 (1)**

Data Characteristics: 4

Forest fires can be detected by several methods, both on a spatial and spectral base. Spatial methods include change detection and pattern recognition. The most promising method for forest fire detection on raw data is currently thermal hotspot detection, which relies on the difference in thermal bands between a target pixel and its background pixel to identify fire activity. [301]. Still, no constant outline or spatial pattern can be set, as wildfire shapes are not constant. Table A.4 summarizes the scores for data characteristics, where the overall criterion scores a 1.

Table A.4: Sub-criteria scores for data characteristics for Wildfire Detection.

Criterion	Justification	Score	Numerical Contribution
No spectral dependency	Average, optical methods exist, but thermal bands are still dominant and the most effective	o	1.25
Clear spatial pattern	No, fire spread patterns are irregular and hard to generalize	-	0
		Total	**1.25 (1)**

Dataset Availability: 5

Wildfire detection is supported by a many publicly available datasets. Both high-resolution cue and low-resolution, wide-FOV tip imagery exist, often accompanied by georeferencing and labels or segmentation masks. Main examples include: [302]

- Active Fire Detection (LANDSAT-8, 30m, 150k images, segmentation, multispectral, georeferenced, zoomed in patches)
- Wildfire Prediction Dataset (Various, 350 x 350 pix, 42.8k images, RGB, balanced fire / no fire)
- Next Day Wildfire Spread (MODIS/VIIRS, 1km, 37k images)
- Forest Fire from HIMAWARI-8 (0.5-1km, wildfire/no-wildfire)
- MOD14A1 (MODIS, 1km, fire mask segmentation, daily, temporal data)

Example patches of the Active Fire Dataset in near-infrared spectrum are shown in Figure A.1, and from the Wildfire Prediction dataset in RGB in Figure A.2.

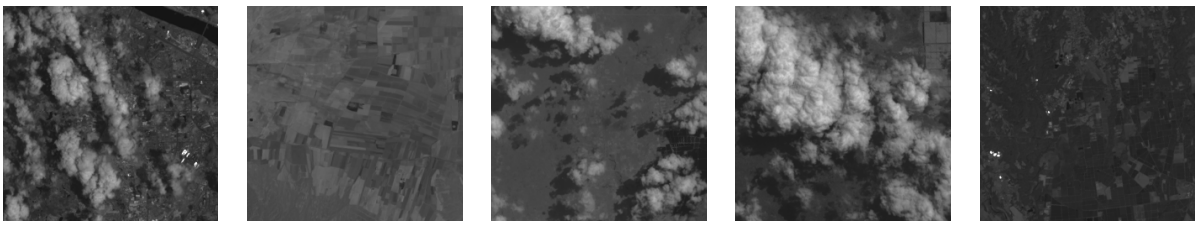


Figure A.1: Example patches of the Active Fire Dataset in near-infrared spectrum [213].

Figure A.2: Example patches of the Wildfire Prediction Dataset in the visual spectrum [303].

Many datasets include multispectral or temporal metadata, are AI-ready, and are openly accessible via Kaggle or government platforms. The only limitation is that most training data reflects late-stage fires with visible flames or smoke, which reduces relevance for early detection [304]. Nevertheless, the size, labels, multispectral nature, and diverse resolution and field-of-view availability justify a maximum score of 5 as shown in Table A.5

Table A.5: Sub-criteria scores for Data Availability for Wildfire Detection.

Criterion	Justification	Score	Numerical Contribution
Tip / wide-FOV data available	Yes, MODIS, VIIRS, etc. >0.5 km/pix	+	1.00
Cue / high-res data available	Yes, LANDSAT, Sentinel < 20 m/pix	+	1.00
Raw data or .tiff formats	Yes, widely available and standardized	+	1.00
Labelled	Yes, most datasets include labels or segmentation masks	+	1.00
Temporal dataset	Yes, daily or hourly datasets exist (e.g., MOD14A1)	+	1.00
		Total	**5.00 (5)**

A.2.2. Permanently Clouded Region Mapping

Tip and Cue Benefit: 4

Mapping permanently clouded regions aligns well with the Tip and Cue system. A large field of view is required to detect gaps in cloud coverage over vast areas. High spatial resolution is needed to obtain high quality data when these gaps appear. While low latency is not critical for this application, high temporal resolution is beneficial to capture more data over time and increasing the chances of observing these gaps. Long revisit times further raise the likelihood of successful observations, and multi-angle views allow better visibility of areas obscured by clouds. Based on these factors, the Tip and Cue system benefit scores 4 for this application as highlighted in Table A.6.

Table A.6: Sub-criteria scores for Tip and Cue benefit for Permanently Clouded Region Mapping.

Criterion	Justification	Score	Numerical Contribution
Large field of view	Yes, needed to identify gaps in the clouds across a vast area	+	0.83
High spatial resolution	Yes, required to obtain rare high-resolution data of frequently covered regions	+	0.83
Low latency	No, there is no time-critical component, real-time response is not really needed	-	0
High temporal resolution	Yes, frequent observations increase chances of capturing cloud breaks	+	0.83
Long revisit time	Yes, long revisit times enable longer observations	+	0.83
Multi-angle view	Yes, alternative angles allow for visibility at different locations	+	0.83
		Total	**4.16 (4)**

Urgency: 4

There is no direct societal urgency associated with clouded region mapping, but it indirectly limits many applications that rely on optical Earth observation. For example, in tropical forests and polar areas, LANDSAT and Sentinel-2 data is unusable for over 80% of the year. El Danubio, Svalbard, and Chuuk Lagoon have annual cloud cover rates of 98.6%, 96.7%, and 92%, respectively [202]. On average, LANDSAT cannot optically monitor regions in Brazil or Indonesia for more than 1-2 months per year [203]. There is no current system that actively exploits cloud gaps in a coordinated manner. Most studies rely on post-processing and cloud compositing, or use SAR data as a substitute. This application is novel, not addressed by existing constellations, and has no defined detection benchmarks. Although cloud mask studies achieve moderate accuracies of 0.8 - 0.9, none of these is focussed at detecting small holes in the clouds [305], Therefore, the urgency criterion scores 4 (see Table A.7).

Table A.7: Sub-criteria scores for Tip and Cue Urgency for Permanently Clouded Region Mapping.

Criterion	Justification	Score	Numerical Contribution
Tip and Cue system need	Yes, there is no current system that addresses this specific use case	+	1.67
Novelty	Yes, very few studies or applications target this concept	+	1.67
Currently achieved performance	Average, cloud masking exists, but not targeting small holes in the clouds	o	0.83
		Total	**4.17 (4)**

Data Characteristics: 3
The collection of data through cloud gaps does not rely on narrow spectral bands or specific wavelengths. Standard optical sensors are sufficient, and there is low sensitivity to off-nadir angles. This enables flexibility in Cue satellite positioning. However, the holes in cloud decks do not follow consistent spatial patterns, making on-board identification more difficult. Prediction of hole location is also non-deterministic, complicating edge processing or prioritization. In addition, while the Cue satellite arrives later at the same location, when the clouds have progressed, an accurate cloud dynamics model would be needed to model clouddeck changes which is not feasible in this first demonstration. Therefore, while spectral independence is high, and spatial consistency low, the data characteristics criterion scores a 3 as summarized in Table A.8.

Table A.8: Sub-criteria scores for Data Characteristics for Permanently Clouded Region Mapping.

Criterion	Justification	Score	Numerical Contribution
No spectral dependency	Ye, standard optical imagery is sufficient, no need for special sensors	+	2.50
Clear spatial pattern	No, cloud gaps lack consistent patterns, difficult to determine the pattern from tipping	-	0
		Total	**2.50 (3)**

Data Availability: 5
Multiple open-access datasets are available for cloud classification and segmentation, both at high spatial resolution and large field of view, with spatial resolutions ranging from 10 m to 2 km. Temporal sequences and georeferenced metadata are available in most datasets. Both tipping and cueing datasets contain segmentation masks for labelling, with the main cloud dataset examples being:

- MODIS Cloud Mask (MOD35) (MODIS, 1 km and 250 m, global coverage, multispectral, cloud classification flags, georeferenced, Level 2 product)
- Sentinel-2 Cloud Mask Catalogue (Sentinel-2, 20 m, 513 subscenes of 1022 x 1022 pixels, cloud segmentation masks, multispectral, georeferenced)
- Himawari-8 Cloud Feature Dataset (HCFD) (Himawari-8, 2 km, East Asia and Western North Pacific, 2016-2022, cloud feature descriptions, multispectral, georeferenced)
- CloudSEN12 (Sentinel-2 and Sentinel-1, 10 m, 49,400 patches of 5090 x 5090 m, cloud and shadow annotations, multispectral and SAR, georeferenced)
- Landsat 8 Cloud Cover Assessment Validation Data (LANDSAT-8, 30 m, 96 scenes across 12 biomes, manually generated cloud masks, multispectral, georeferenced)
- 95-Cloud (Landsat 8, 384x384 pixel patches, 10m, 34,701 training images, 9,201 test images, RGB and NIR bands, binary cloud segmentation, georeferenced)

Example patches of the 95-Cloud dataset with its mask are shown in Figure A.3, visualizing the irregular patterns in the gaps between the clouds.

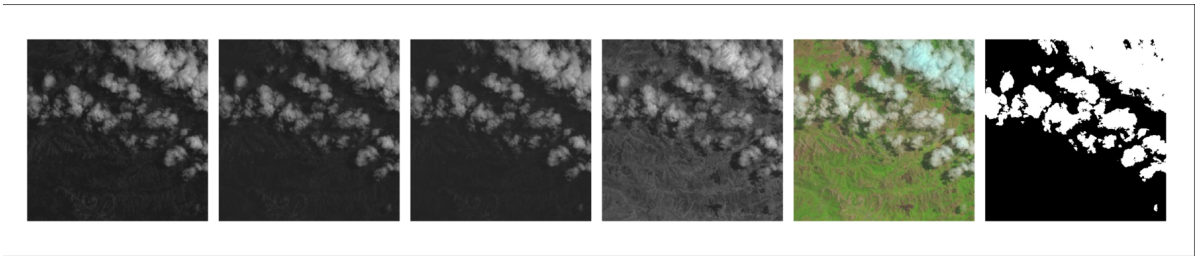


Figure A.3: Example patches of the 95-Clouds dataset. From left to right: Red, Green, Blue, NIR, false color, cloud mask. [306]

The availability of high and low resolution, labelled, raw, and temporal data fully meets the criteria, and therefore achieves the maximum score of 5 as shown in Table A.9.

Table A.9: Sub-criteria scores for Data Availability for Permanently Clouded Region Mapping.

Criterion	Justification	Score	Numerical Contribution
Tip / wide-FOV data available	Yes, MODIS (MOD35) at 1 km & 250 m, Himawari-8 at 2 km provide wide field of view data	+	1.00
Cue / high-res data available	Yes, Sentinel-2 at 10-20 m, Landsat-8 at 30 m; datasets like CloudSEN12 and 95-Cloud offer labeled patches	+	1.00
Raw data / .tiff formats	Yes, Georeferenced .tiff formats available in MODIS, Sentinel-2, and Landsat-8 datasets	+	1.00
Labelled	Yes, major datasets include cloud masks or binary labels	+	1.00
Temporal dataset	Yes, Himawari-8 (2016-2022), MODIS (daily), CloudSEN12 (multi-temporal patches)	+	1.00
		Total	**5.00 (5)**

A.2.3. Illegal Martime Activities

Tip and Cue Benefit: 5

Illegal maritime activity detection benefits from all Tip and Cue features. A large field of view is necessary to scan open seas for unreported or hidden vessels. High spatial resolution is required to distinguish small vessels and accurately identify the ships. Low latency is important for real-time response and enforcement. High temporal resolution supports the detection of unusual movement patterns, including loitering or course anomalies. Long revisit times enable multi-day tracking and historical comparison. Multi-angle views can help to identify the vessel from multiple angles, improving identification accuracy. Because all the subcriteria are satisfied, this criterion scores 5 as summarized in Table A.10.

Table A.10: Sub-criteria scores for Tip and Cue benefit for Illegal Maritime Activities.

Criterion	Justification	Score	Numerical Contribution
Large field of view	Yes, required to scan vast ocean regions	+	0.83
High spatial resolution	Yes, needed to distinguish vessel types and detect illicit behavior	+	0.83
Low latency	Yes, critical for fast enforcement or interception	+	0.83
High temporal resolution	Yes, enables pattern detection over time (e.g., loitering, AIS spoofing)	+	0.83
Long revisit time	Yes, enables longer comparison to AIS data and pattern recognition	+	0.83
Multi-angle view	Yes, supports detailed vessel profiling from various perspectives	+	0.83
		Total	**5.00 (5)**

Urgency: 2

Dark vessel detection is an active field with increasing societal interest. Several commercial constellations already serve this purpose. ICEYE provides SAR data at 0.5 m resolution and 5 km swath. PlanetScope operates 130 optical satellites at 3 m resolution and 32 km swath with daily revisit. SkySat provides high-resolution imagery (50 cm) up to 10 times per day with a 6 km swath [307], [308]. These systems already support maritime surveillance, often using manual tasking and AIS correlation. Tip and Cue would automate and optimize this workflow, but it does not introduce a fully new capability. Still, vessel classification performances remain moderate [309], because a vessels appearance can significantly vary with lighting condition, viewing geometry, and seastate, and variation between ships from the same class [310]. Summing up, the urgency scores a 2 as shown in Table A.11.

Table A.11: Sub-criteria scores for Tip and Cue Urgency for Illegal Maritime Activities.

Criterion	Justification	Score	Numerical Contribution
Tip and Cue system need	Maybe, can also be fulfilled by tasking high resolution satellites (like SkySat and PlanetScope)	o	0.83
Novelty	No, maritime monitoring is an established domain	-	0
Currently achieved performance	Average, current algorithms perform well on existing datasets, but classification performance can be enhanced	o	0.83
		Total	**1.66 (2)**

Data Characteristics: 5

Ship detection is not dependent on spectral bands, as they are identifiable in SAR, RF, and optical imagery. The ship body is a consistent spatial pattern which is ideal for segmentation or classification models. These patterns remain visible under various viewing angles, satisfying both subcriteria. Therefore, data characteristics scores a 5 for illegal maritime activity detection, as highlighted in Table A.12.

Table A.12: Sub-criteria scores for Data Characteristics for Illegal Maritime Activities.

Criterion	Justification	Score	Numerical Contribution
No spectral dependency	Yes, detection possible with SAR, RF, or optical imagery	+	2.50
Clear spatial pattern	Yes, ships have structured, repeatable geometry; ideal for spatial models	+	2.50
		Total	**5.00 (5)**

Data Availability: 4
There is broad dataset availability for both low- and high-resolution imagery. Datasets like Airbus Ship, HRSC2016, DIOR, xView, FGSD, and ShipRSImageNet offer high-res images with annotations for multiple vessel types. For tipping, S2-SHIPS and LEVIR-Ships offer lower resolution and wider field of view. SAR data is also available, including xView3-SAR with AIS correlation. Main dataset examples include: [302]

- Airbus Ship (Airbus SPOT, 0.3-6 m, 192,556 images, rotated bounding boxes, large-scale, widely used in detection benchmarks)
- xView (WorldView, 0.3 m, 1,414 images, horizontal bounding boxes, 9 ship classes, georeferenced, high-resolution, AI-ready)
- HRSC2016 (Google Earth, 0.4-2 m, 1,061 images, hbox/rbox/seg, 25 ship classes, widely cited in ship classification studies)
- DIOR (Google Earth, 0.5-30 m, 23,463 images, hbox annotations, multi-class dataset, large-scale, benchmark-ready)
- ShipRSImageNet (Multiple sources, 0.12-6 m, 3,435 images, hbox/rbox/seg, 50 ship classes, AI-ready, used for fine-grained classification)
- xView3-SAR (Sentinel-1, 20 m, 991 images, hbox, SAR data for dark vessel detection, includes AIS correlation, benchmark dataset)
- S2-SHIPS (Sentinel-2, 10 m, 16 scenes, hbox and segmentation, low-res wide-FOV data, suitable for tipping stage)
- VHRShips (Google Earth, 0.43 m, 6,312 images, hbox annotations, 34 ship types, good spatial resolution for cueing)
- FGSD (Google Earth, 0.12 -1.93 m, 2,612 images, rotated boxes, 43 ship classes, used for complex multi-class detection)
- LEVIR-Ships (GF-1 and GF-6, 16 m, 3,896 images, horizontal boxes, single-class ship detection, used for long-range and wide-area studies)
- SRSDD-v1.0 (GF-3 Spotlight SAR, 1 m, 30 SAR images, 2,884 ships, 6 categories, 1024×1024 pixels, challenging inshore clutter)

Example patches of the LEVIR-Ships, UOW-Vessel, and SRSDD dataset are shown in Figure A.4, Figure A.5, and Figure A.6 respectively, to highlight a variety of Tip, Cue, optical, and SAR data.

Figure A.4: Example patches of the LEVIR-Ships Dataset in the visual spectrum [311].

Figure A.5: Example patches of the UOW-Vessel Dataset in the visual spectrum [312].

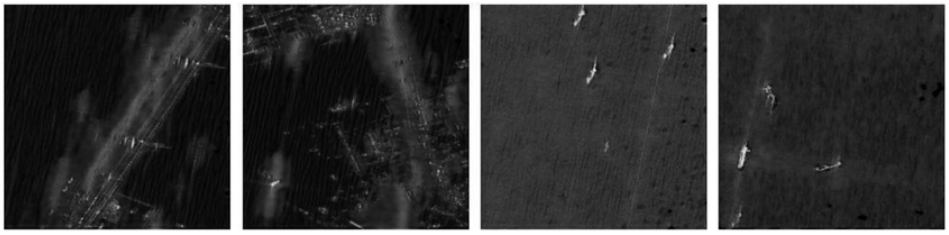


Figure A.6: Example patches of the SRSDD Dataset in SAR spectrum [313].

The datasets are generally raw, labelled, and AI-ready. However, temporal datasets with consistent time-series or AIS integration are limited. This can be a critical limitation, because dark vessel detection often requires temporal correlation with AIS data. The total score for data availability is a 4 as shown in Table A.13.

Table A.13: Sub-criteria scores for Data Availability for Illegal Maritime Activities.

Criterion	Justification	Score	Numerical Contribution
Tip / wide-FOV data available	Yes, medium, high field of view datasets are available, particularly SAR	+	1.00
Cue / high-res data available	Yes, large archive of high-res ship datasets	+	1.00
Raw data / .tiff formats	Yes, multispectral and .tiff available	+	1.00
Labelled	Yes, extensive annotations across public and commercial datasets, even per ship	+	1.00
Temporal dataset	No, typically absent and no fused AIS dataset available	-	0
		Total	**4.00 (4)**

A.2.4. Wildlife Monitoring

Tip and Cue Relevance: 4

Wildlife tracking requires a wide field of view to monitor animal populations in vast land scapes. A very high spatial resolution is needed to distinguish different animals, particularly to classify in which species group an individual belongs to. It should be noted that wildlife surveys are typically conducted during specific phases of the annual population cycle, where a high revisit rate helps to ensure imagery availability during these periods. A high temporal resolution and long revisit time increases data for tracking pattern and behavior monitoring, and multi-angle view can further enhance species identification. Table A.14 summarizes the characteristics, summing to a criterion score of 4.

Table A.14: Sub-criteria scores for Tip and Cue benefit for Wildlife Monitoring.

Criterion	Justification	Score	Numerical Contribution
Large field of view	Yes, needed for monitoring vast oceans or landscapes	+	0.83
High spatial resolution	Yes, required to distinguish individuals and species	+	0.83
Low latency	No, no urgent response needed; seasonal monitoring windows are sufficient	-	0
High temporal resolution	Yes, many samples give higher certainty in tracking patterns	+	0.83
Long revisit time	Yes, useful for wildlife tracking patterns and behavior observation	+	0.83
Multi-angle view	Yes, helps with species identification	+	0.83
		Total	**4.15 (4)**

Urgency: 5

Although not directly tied to high societal urgency, the need for improved wildlife monitoring capabilities

is growing. Many populations are endangered or understudied, especially in remote regions. Also, the presense of wildlife is often an indicator for ecosystem health, therefore is becoming increasingly important in Earth conversation efforts [314]. Current commercial satellite tasking is expensive and inefficient, particularly due to uncertainty in animal locations. This makes the purchase of data unfeasible and expensive for many researchers, limiting monitoring capabilities. Tip and Cue would reduce data cost by only cueing when wildlife is detected. The application is still emerging, and most detection models rely on manual review. A large portion of published work uses manual annotation pipelines due to poor automated performance. The field is novel, with low operational maturity and poor dataset support in many regions, and therefore scores a 5 as shown in Table A.15.

Table A.15: Sub-criteria scores for Tip and Cue application need for Wildlife Monitoring.

Criterion	Justification	Score	Numerical Contribution
Tip and Cue system need	Yes, early-stage application with poor automated performance	+	1.67
Novelty	Yes, still an emerging domain due to low detection likelihood	+	1.67
Currently achieved performance	Low in practise, often require manual review	+	1.67
		Total	**5.00 (5)**

Data Characteristics: 5

Wildlife detection does not depend on narrow spectral bands. Detection is possible through contrast, shape, and thermal anomalies. Species like whales, elephants, and penguins are consistently detectable via spatial features which also remain detectable at moderate off-nadir angles. Both subcriteria of low spectral dependency and clear spatial pattern are fully met as displayed in Table A.16, summing up to 5 as the total score.

Table A.16: Sub-criteria scores for Data Characteristics for Wildlife Detection.

Criterion	Justification	Score	Numerical Contribution
No spectral dependency	Yes, detection based on shape, contrast, and temperature not narrowband imaging	+	2.50
Clear spatial pattern	Yes, animals exhibit consistent geometric features across environments	+	2.50
		Total	**5.00 (5)**

Data Availability: 3

Considering data availability, most the most commonly used technologies for wildlife monitoring include WorldView, with a spatial resolution of 31cm, swath width of 13.1km, and an average revisit time of less than a day upon request [315]. Followed by QuickBird, with 60cm spatial resolution at 16.5km swath width and revisit time of every 2-3 days. And GeoEye with 40cm spatial resolution, 15.2km swath width, and revisiting every 2-5 days. The same paper demonstrates that a resolution of at least 0.5m / pixel is required for the tipping satellite to distinguish large animals from space [42]. However, public datasets of high swath remote sensing wildlife images are barely available. [316] outlined the wildlife tracking datasets available, showing that whales, birds, elephants, and penguins have the highest data availability. Upon further inspection, UAV and high resolution satellite datasets exist for each species: elephants [315] [317], penguins [318] [319] [320], whales [100] [96], but low resolution, wide field of view data shall probably be created through downsampling full VHR satellite images. So far, temporal datasets, georeferenced, or .tiff formats are barely available. Given most wildlife datasets are available for whales, an overview of these is given below [43] and Table A.17 summarizes the final criterion score of 3.

Example patches from the Whales from Space dataset are given in Figure A.7. Example patches of wide field of view data are shown in Figure A.8, however it should be noted that the full dataset is for purchase.

Figure A.7: Example patches of the Whales from Space Dataset in the visual spectrum [100]. *Satellite image © 2022 Maxar Technologies.*

Figure A.8: Example patches of commercial wide field of view data of whales compared to ships in the visual spectrum. From left to right: 4 whales, 1 boat [321]; three whales [91]; one boat [91]

- Whales from Space (WorldView-2, 0.5 m, multispectral, southern right whales)
- CNN-Whales-from-Space (Google Earth, 0.15-2.5 m, multispectral and panchromatic, whales)
- Borowicz Whale Dataset (WorldView-3, 0.31 m, pansharpened multispectral, whales, aerial survey comparison, ResNet and DenseNet)
- Bamford Baleen Dataset (WorldView-3, 0.31 m, panchromatic, baleen whales, manual screening)
- Platonov Marine Mammal Dataset (GeoEye-1, 0.5 m, panchromatic, bowhead whales, walruses, and polar bears)
- Cubaynes Whale Dataset (WorldView-3, 0.31 m, pansharpened multispectral, gray, southern right, and fin whales)
- Albiez Humpback Dataset (IKONOS, 1 m, multispectral, humpback whales)
- Thums Whale Dataset (WorldView-2 and 3, 0.5 m and 0.31 m, pansharpened multispectral, humpback whales)
- Charry Arctic Whale Dataset (WorldView-3, 0.31 m, panchromatic, beluga and narwhal whales)
- Fretwell Baleen Dataset (WorldView-2, 0.5 m, multispectral, baleen whales)

Table A.17: Sub-criteria scores for Data Availability for Wildlife Detection.

Criterion	Justification	Score	Numerical Contribution
Tip / wide-FOV data available	Maybe, generally missing but commercial data can be converted	o	0.50
Cue / high-res data available	Yes, many sources exist (WorldView, QuickBird), might be commercial	+	1.00
Raw data / .tiff formats	No, rarely available in open or standardized formats	-	0
Labelled	Yes, most datasets include bounding boxes, manual annotations, or segmentations	+	1.00
Temporal dataset	No, temporal data is limited or nonexistent in most studies	-	0
		Total	**2.50 (3)**

[0]P. Fretwell, "Counting whales by satellite," *British Antarctic Survey*, Mar. 21, 2017. Available: `https://blog.maxar.com/earth-intelligence/2017/counting-whales-by-satellite`. Accessed: Apr. 10, 2025.

A.2.5. Application Trade-off

Summing up all the criteria scores, Table A.18 shows the results of the application trade-off. From this, wildlife conversation comes out as the winner, followed closely by the runner ups of permanently clouded region mapping and illegal maritime activity detection. Wildfire detection reaches last place, due to low urgency and lack of spatial pattern.

Table A.18: Application trade-off based on Tip and Cue benefits, urgency, data characteristics, and data availability. Sub-scores of -, o, and + are given for the sub-criteria, which are converted to scores between 0 and 5 for each criteria through weighted summation.

Criterion	Sub-criterion / Criterion Score	Wildfires	Permanently Clouded Region Mapping	Illegal Maritime Activities	Wildlife Conservation (whales)
Tip and Cue Benefit	Large Field of View	+	+	+	+
	High Spatial Resolution	+	+	+	+
	Low Latency	+	-	+	-
	High Temporal Resolution	+	+	+	+
	Long Revisit Time	+	+	+	+
	Multi-angle View	-	+	+	+
	Criterion Score	**4**	**4**	**5**	**4**
Urgency	Tip and Cue System Need	o	+	o	+
	Novelty	-	o	-	+
	Currently Achieved Performance	-	+	o	+
	Criterion Score	**1**	**4**	**2**	**5**
Data Characteristics	No Spectral Information	o	+	+	+
	Clear Spatial Pattern	-	-	+	+
	Criterion Score	**1**	**3**	**5**	**5**
Data Availability	Tip / High FOV Data Available	+	+	+	o
	Cue / High res Data Available	+	+	+	+
	Raw Data / Multispectal Available	+	+	+	o
	Labelled Dataset	+	+	+	+
	Temporal Dataset	+	+	-	-
	Criterion Score	**5**	**5**	**4**	**3**
	Total Score	**11**	**16**	**16**	**17**

A.2.6. Trade-off Sensitivity Analysis

The trade-off shows that wildlife monitoring (17), permanently clouded region mapping (16), and illegal maritime activity detection (16) all score within a one-point range. This means that changing one sub-score, adding or removing a subcriterion, adjusting the weights, or even changing the rounding strategy could already lead to a different outcome, making the trade-off not decisive. When considering the non-rounded scores: 16.65, 15.83, and 15.66 respectively, still reflecting the same winner, it also demonstrates how marginal these differences are.

This makes clear that the current scoring system does not provide a definitive outcome and the trade-off is not robust on its own. The ranking is highly sensitive to minor adjustments. For example, changing the temporal dataset score of wildlife from '-' to 'o' or interpreting urgency of illegal maritime detection slightly differently would reorder the top 3.

One way to improve certainty would be to change the method by using different weights, adding new criteria like implementation complexity or system readiness, or by scoring each subcriterion on a wider numerical scale. However, as the scoring system and criteria were fixed from the beginning, changing them afterwards would compromise the fairness and comparability of the evaluation. Therefore, this was not implemented.

Instead, reasoning was used to reach the final conclusion. Permanently clouded region mapping, while balanced in scores, loses on spatial pattern. This would require a complex tipping strategy and cloud dynamics modelling, which shall not be the focus of the first demonstration. Given the limited time and aim for a more general use case, its exclusion was justified.

Illegal maritime activity detection loses primarily on urgency. While it is an active field, many capabilities already exist operationally. Moreover, the lack of temporal and AIS-fused training data means the core output: distinguishing legal from illegal activity, cannot be validated. Without this, the use case turns into generic vessel detection, which is already being done. It would become a technology demo without a practical outcome.

Wildlife monitoring mainly loses points on data availability, especially for the Tip satellite. However, it

remains a highly novel and early-stage application. It is not well covered by existing systems, and detection still relies on manual work. That makes it the most interesting use case for this AI-based Tip and Cue demonstration project when feasibility and impact are both considered. Because the main limiting factor, the availability of Tip data, has to be confirmed through a more in depth research, it was decided to work out whale detection in the second part of this literature review.

It was also considered whether wildlife and maritime vessel detection could be combined into a single use case, as both involve small objects on the ocean surface and rely on similar high-resolution satellite data. In practice, whales must be distinguished not only from the ocean background, but also from vessels and other marine objects, which makes this distinction operationally relevant.

However, the objective of this thesis was to demonstrate the AI-based Tip and Cue framework under controlled and well-defined conditions. Introducing vessel data would have shifted the focus toward multi-class maritime classification, requiring additional datasets, temporal correlation, and operational context beyond the scope of this demonstration for offnadir effect quantification. Therefore, the study was deliberately limited to whale detection as a single-class target.

B

Appendix B - AI-Based Tip and Cue for Whale Monitoring

This appendix summarizes the whale-monitoring context that informed the final thesis setup. It compiles the satellite systems, datasets, and detection approaches identified in the literature review for whale monitoring, with a specific focus on their relevance for an AI-Based Tip and Cue architecture. The appendix serves as supporting background for the choice of whale monitoring as demonstration use case and for the benchmark assumptions adopted in the main report.

B.1. Tip Satellites for Whale Detection

Table B.1: Overview of satellites for tipping, with a spatial resolution of 5 meters or higher.

Satellite	Satellite Company	Launch Date	Expected EOL	Panchromatic Resolution	Multispectral Resolution	Swath Width	Revisit Rate	Spectral Bands
Sentinel-2A	European Space Agency	23 Jun 2015	Jul 2025	-	10/20/60 m	290 km	5 days	13 bands: RGB + NIR (10 m), SWIR (20 m) + atmospheric (60 m)
Sentinel-2B	European Space Agency	07 Mar 2017	>2027	-	10/20/60 m	290 km	5 days	13 bands: RGB + NIR (10 m), SWIR (20 m) + atmospheric (60 m)
Landsat-8	NASA	11 Feb 2013	Feb 2028	15 m	30/100 m	185 km	8 days	11 bands: RGB + VNIR, SWIR, Thermal (100 m), Panchromatic
Landsat-9	NASA	27 Sep 2021	Sep 2031	15 m	30/100 m	185 km	8 days	11 bands: RGB + VNIR, SWIR, Thermal (100 m), Panchromatic
RapidEye	Planet Labs	29 August 2008	31 March 2020	-	6.5 m	77 km	1 day	5 bands: Blue, Green, Red, Red-Edge, NIR

B.2. Cue Satellites for Whale Detection

Table B.2: Overview of VHR satellites available for cueing, with a spatial resolution of 50 cm or lower.

Satellite	Satellite Company	Launch Date	Expected EOL	Panchromatic Resolution	Multispectral Resolution	Swath Width	Revisit Rate	Spectral Bands
IKONOS-2	Maxar Technologies	24 Sep 1999	31 Mar 2015	1 m	4 m	11 km	1 day	5 bands: Blue, Green, Red, NIR, Panchromatic
QuickBird-2	Maxar Technologies	18 Oct 2001	17 Dec 2015	61 cm	2.4 m	16.5 km	1.5 days	5 bands: Blue, Green, Red, NIR, Panchromatic
WorldView-1	Maxar Technologies	18 Sep 2007	Sep 2025	50 cm	-	17.6 km	1.7 days	1 band: Panchromatic
GeoEye-1	Maxar Technologies	06 Sep 2008	Sep 2025	41 cm	1.64 m	15.2 km	<3 days	5 bands: Blue, Green, Red, NIR, Panchromatic
WorldView-2	Maxar Technologies	08 Oct 2009	Dec 2025	46 cm	1.84 m	16.4 km	1.1 days	9 bands: Coastal, Blue, Green, Yellow, Red, Red-Edge, NIR1, NIR2, Panchromatic
Pleiades-1A	Airbus	16 Dec 2011	Dec 2026	50 cm	2 m	20 km	1 day	5 bands: Blue, Green, Red, NIR, Panchromatic
Pleiades-1B	Airbus	01 Dec 2012	Dec 2026	50 cm	2 m	20 km	1 day	5 bands: Blue, Green, Red, NIR, Panchromatic
WorldView-3	Maxar Technologies	13 Aug 2014	Aug 2025	31 cm	1.24 m	13.1 km	< 24 hours	29 bands: 8 VNIR (incl. RGB), 8 SWIR, 12 CAVIS, Panchromatic
WorldView-4	Maxar Technologies	11 Nov 2016	07 Jan 2019	31 cm	1.24 m	13.1 km	< 24 hours	5 bands: Blue, Green, Red, NIR, Panchromatic
PlanetScope Constellation	Planet Labs	22 Jun 2016	> 2028	-	3.7 m	24 km	1 day	8 bands: Red-Edge, Red, Green, Green I, Yellow, Blue, Coastal Blue, NIR, Panchromatic
Pleiades Neo 3	Airbus	29 Apr 2021	Apr 2031	30 cm	1.2 m	14 km	12 hours	7 bands: Deep Blue, Blue, Green, Red, Red-Edge, NIR, Panchromatic
Pleiades Neo 4	Airbus	16 Aug 2021	> 2031	30 cm	1.2 m	14 km	12 hours	7 bands: Deep Blue, Blue, Green, Red, Red-Edge, NIR, Panchromatic
SuperView Neo-1	SuperView	29 Apr 2022	Apr 2028	30 cm	1.2 m	12 km	12 hours	5 bands: Blue, Green, Red, NIR, Panchromatic
EOS SAT-1	EOS Data Analytics	03 Jan 2023	2028 - 2030	1.4 m	2.8 m	22 km	1 day	11 bands: RGB, 2 NIR, 3 Red-Edge, Watervapor, Aerosol, Panchromatic
Pelican-1	Planet Labs	11 Nov 2023	Dec 2028	30 cm	1.2 m	6.5 km	30 min	6 bands: Blue, Green, Red, Red Edge, Near-Infrared, Panchromatic
WorldView Legion	Maxar Technologies	02 May 2024	Sep 2032	29 cm	1.16 m	9 km	1.6 hours	9 bands: Coastal Blue, Blue, Green, Yellow, Red, Red Edge 1, Red Edge 2, Near-Infrared (NIR), Panchromatic
Pelican-2	Planet Labs	14 Jan 2025	> 2030	40 cm	50 cm	12 km	< 1 hour	6 bands: Blue, Green, Red, NIR, SWIR1, SWIR2
Clarity-1	Albedo	17 Mar 2025	> 2026	10 cm / 2m	2 m	3.5 km	4.8 hours	5 bands: RGB (10 cm), Thermal IR (2m), Panchromatic
EOI-1	EOI Space	> 2025	> 2030	15 cm	1 m	5 -7 km	45 min	5 bands: Blue, Green, Red, NIR, Panchromatic

B.3. Whale Monitoring Datasets

Table B.3: VHR whale datasets overview, including species, satellite platforms, and access information. Sorted by availability.

Study	Species	Satellite Platform	Resolution, format	Number of images, area	Number of whales	Labeling	Access / data availability
[100]	Southern right whale, Humpback whale, Fin whale, Grey whale	WorldView-3, WorldView-2, GeoEye-1, Quickbird-2	0.5m, .png	9 satellite images, 634 patches, 128 x 128 pix, 6300 km^2	633 whales, ≈1 whale patch	Labelled with point or box shapes	Patches available upon request: PDCServiceDesk@bas.ac.uk. Full image ID's in table 1.
[101]	Synthetic whales, comparable to Whales from Space	Synthetic images through SeaDroneSim2 Simulator	140 x 140 - 30,000 x 30,000 pix, .png	2000 images	One to multiple whales per image	Segmentation masks	Available upon request: https://github.com/prgu
[43]	Various species	DigitalGlobe archive, Google Earth Pro	.png	6000 patches, 1500 original + 663 whales in space + 3837 augmented, cropped to 416 x 416 pix	1 whale per image	Labelled with whale and probable whale	Available upon request. 20 patches at: https://github.com/SaakshiKapoor/whales/tree/SaakshiKapoor-patch-1
[322]	Beluga whale	WorldView-2, WorldView-3, GeoEye-1	0.31-0.41m, .tiff	11 WV-3 images, 3,842 km^2, 230,967 patches of 250 x 250m.	10 639 whales	Labelled by crowd-counters with points and bounding boxes	Available upon request
[323]	Gray whales	WorldView-3	0.31m, .tiff	Three WV-3 image sections	7 + 8 + 12 definite	Labelled with classes: whale, water, objects; 202 features in total	Available upon request
[11]	Various species	Google Earth, Arkive, NOAA images	0.15 - 2.5m, .png	700 VHR images + 945 Arkive and NOAA images	68 whales, 400 ships, 400 submerged rocks	Labelled with species ID	Available upon request
[324]	Beluga whales	Aerial survey from DFO	0.1m, .png	467 images, 4912 x 7360 pix	Not specified	Labeled with whale coordinates from expert annotators	Available upon request
[92]	Blue whales, Humpback whales, Antarctic minke whales	Aerial survey	30-80m altitude, 6000 x 4000 pix, .jpg	326 + 58 images	1-2 whales per image	Labelled with masks and species ID	https://doi.org/10.5061/dryad.7482v2n
[93]	Beluga whale	Aerial survey from DFO	.jpg	5,334 aerial images, 875m x 585m each	4572 whales	Manual segmentations by three observers	Available upon request: cortney.watt@dfo-mpo.gc.ca.
[89]	Southern right whales	WorldView-3	0.31m, .tiff	Two WV-3 images, ≈400 km^2	42 whales	Not specified	WV-3 for purchase, ID: 1040010003121A00, 1040010029924200
	Southern right whales, Humpback whales	Aerial survey	0.02m, .png (downsampled to 0.31m)	194 aerial images, observed area not specified	17 whales	Not specified	Aerial images downsampled: https://zenodo.org/records/3356970
[91]	Gray whales	WorldView-2, WorldView-3, GeoEye-1	0.31-0.5m, .tiff, .png	4 satellite images, 313 patches	503 whales, 103 boats	Labelled with bounding boxes around whales and boats	For purchase: ID: 103001001E12C300, 103001003C8B2300, 10400100295ED00, 1050410001FEA500
		Aerial survey	0.005m, .jpg	Two aerial image datasets	1124 instances, 53 unique animals	Labelled with bounding boxes	Not specified
[98]	Baleen whales	WorldView-3	0.31m, .tiff	Four WV-3 images, 866 km^2 total	18 definite, 21 probable, 146 unclassified	Not specified	For purchase, ID: 1040010038AFA900, 1040010039052600, 10400100375 7E500, 104001003AC95B00
[325]	Beluga, Narwhal whales	WorldView-3	0.31m, .tiff	12 WV-3 images, 138.8 km^2 + 127.0 km^2	292 beluga whales, 109 narwhals	Excel with latitude, longitude, and time	For purchase, ID's at: https://doi.org/10.1371/journal.pone.0254380.s001
[326]	Beluga whale	WorldView-1, WorldView-2, WorldView-3	0.31-0.5m, .tiff	17 WV images, 9690 km^2	704 certain, 1438 uncertain	Shapefiles available with spatial information	Images for purchase. ID's under Resource identification initiative. Shapefiles available upon request: bryanna.sherbo@dfo-mpo.gc.ca cortney.watt@dfo-mpo.gc.ca
[88]	Southern right whales	WorldView-2	0.5m, .tiff	One WV-2 image, 113 km^2	55 probable 23 possible	Not specified	For purchase, ID 103001001C8C0300
[327]	Humpback whales	WorldView-2	0.5m, .tiff	One WV-2 image, 941 km^2	1 certain 19 probable 33 possible	Not specified	For purchase, ID 10AUG06021738-M2AS-055137769010_01_P0011
[328]	Gray whales	WorldView-2, GeoEye-1	0.5m, .tiff	Three satellite images	Multiple whales per image	Shapefiles provided for whale detection locations	For purchase, ID's Worldview-2: 50103001001E12C300, 50103001003C8B2300 GeoEye-1: 501050410001FEA500 Shapefiles provided
[104]	Baleen whale strandings	WorldView-2	0.5m, .tiff	Two WV-2 images, 836 km^2 + 288 km^2	45 stranded whales	Excel with latitude and longitude	For purchase, ID: 103001003F7A9900, 1030010040C2EE00.
[329]	Beluga whales	WorldView-2, WorldView-3, ESRI World Imagery	0.31-0.5m, .tiff, .png	Two WV images, 152 km^2 + 2,550 km^2.	1147 whales	Not specified	Excel with lat, lon available. For purchase, ID: 103001005AD12400, 104001001D6F2200
[99]	Beluga whales	WorldView-2, WorldView-3	0.31-0.5m, .tiff	Two WV-2 and six WV-3, 3124 km^2, 1281 km^2 of water	393 (2020) + 197 (2021) + 194 (2022) whales	Shapefiles with location	Images not available. Shapefiles available upon request: belang21@myumanitoba.ca or cortney.watt@dfo-mpo.gc.ca
[330]	Beluga whales	WorldView-3	0.31m, .tiff	Two WV-3 images. 192 km^2 + 146 km^2	79 + 75 whales	Not specified	For purchase: ID: 1040010076A6B600, 10410079144500
[331]	Southern right whales	DJI Inspire 2 drone Rapid Eye, PlanetScope, LandSat-8, Sentinel-2, Google Earth	3cm, .png 0.5-30m, .tiff, .png	Two areas, 2.66 + 2.42 km^2 Eight satellite images, cropped to 15.8 km^2 each	71 whales 19 definite, 5 probable, 19 possible	Not specified	Not specified Not specified
[95]	Beluga whales	Aerial surveys from DFO	0.1m, 7360 x 4912 pix, .jpg	6600 images (3888 + 2712)	5328 whales, 6118 non-whales	Not specified	Not specified
[94]	Various species	Aerial images from DFO	7360 x 4920 pix, .jpg	1544 images, mainly water	168 whales, mainly water, 2198 fishing gear, 1652 animal, ≈1000 other	Labelled with ellipses	Not specified
[332]	Southern right whale	Ikonos, GeoEye, WorldView-1, WorldView-2, WorldView-3	0.31-0.81m, .tiff	52 satellite images, 1160 patches, 1,644,060 artificial images	580 whales, 580 non-whales	Not specified	Not specified
[90]	Southern right whale	WorldView-2	0.5m, .tiff	Two WV-2 images, cropped to 800 x 800 pix, converted to .png	125 definite, 65 likely, 127 possible	Not specified	Not specified
[144]	Humpback whales	Thermal rotating line scanner	0.05 - 0.03 deg per pixel, IR	Rotating line scans at four locations	Not specified	Not specified	Not specified
[333]	North Atlantic right whale	Optical satellite imagery	0.15m, .tiff	1 satellite image	25 whales	Not specified	Not specified
[97]	Fin whales	WorldView-3, Google Earth Pro, Aerial, Thermal	<1m, .jpeg,	172 Google Earth Pro + 33 WV-3 + 1208 aerial + 96 thermal images, cropped to 640 x 640 px	≈ 600 whales, 88 ships	Not specified	Not specified

B.4. Whale Monitoring Deep Learning Approaches

Table B.4: Overview of deep learning models used for whale monitoring via aerial and satellite imagery. Metrics include precision, recall, F1-score, and task specific values (mAP, IoU). Only explicitly reported results are shown.

Study	Detection Method	Network Type	Precision	Recall	F1	Other metrics	Code availability
[88]	Thresholding (Band 5)	Detection	0.762	0.846	-	-	-
[89]	ResNet-152	Classification	1.000	0.937	0.968	-	https://github.com/lynch-lab/Borowicz_etal_Spacewhale/tree/1.0.0
	ResNet-18/34	Classification	1.000	0.937	0.968	-	-
	DenseNet	Classification	0.999	0.922	0.959	-	-
[92]	Mask R-CNN, ResNet-101	Segmentation	0.983	1.000	-	0.85 (mIoU)	https://github.com/patrickcgray/cetacean_photogram
[11]	Inception-v3	Classification	-	-	0.818	-	https://github.com/EGuirado/CNN-Whales-from-Space
	Inception-ResNet v2	Detection	-	-	0.94	-	-
[94]	Faster R-CNN (ResNet-50, FPN)	Detection	-	0.555 (AR)	-	0.732 (mAP50)	-
	DeepLabv3	Segmentation	-	-	-	0.4142 (mIoU)	-
[95]	Custom SE-ResNet, ensemble of 10 models	Classification	-	≈0.90–0.95	–	0.63 - 0.79 (PTPVSA)	-
[90]	SPACEWHALE Faster R-CNN	Detection	1.00	1.00	1.00	- -	-
[332]	20 CNN, pre-trained on ImageNet, output feature vector, classify with SVM.	Classification					
	NasNet Large		1.00	1.00	1.000	-	-
	InceptionV3		1.00	1.00	1.000	-	-
	Shufflenet		0.95	1.00	0.974	-	-
	NasNet Mobile		0.95	1.00	0.974	-	-
	SqueezeNet		0.95	1.00	0.974	-	-
	GoogLeNet		0.92	0.65	0.765	-	-
	ResNet-18		0.88	0.70	0.778	-	-
	MobileNetV2		0.88	0.75	0.809	-	-
	ShuffleNet (alt)		0.93	0.70	0.799	-	-
	ResNet-101		0.94	0.75	0.833	-	-
[93]	U-Net + EfficientNet-B3 encoder	Segmentation	0.9226	0.9896	0.9549	-	-
[91]	YOLOv5s (MS-COCO pre-trained)	Detection	0.939	0.843	-	0.884 (mAP50)	https://github.com/KMacfarlaneGreen/yolov5
[43]	Modified Tiny YOLOv3 + ESRGAN	Detection	0.95	0.79	0.86	0.66 (mAP50), 0.664 (IoU50)	https://github.com/SaakshiKapoor/whales
[324]	Faster R-CNN (MS COCO pre-trained, fine-tuned)	Detection	-	-	-	0.79 (mAP50)	-
[101]	U-Net, FPN + SeaDroneSim2	Segmentation	-	-	-	0.746 (DR50), 0.551 (DR60)	https://github.com/prgumd/SeaDroneSim2
[96]	SAM	Segmentation	0.85	0.79	0.82	0.51 (IoU)	-
	YOLO	Detection	0.983	0.977	-	0.992 (mAP50)	-
	EfficientNet	Classification	0.99	0.99	0.79	0.80 (IoU)	-
[97]	YOLOv8 (CSPDarknet + PANet)	Detection	0.87	0.96	-	0.95 (IoU), 0.95 (mAP)	-

B.5. Whale Species

Table B.5: Overview of studied whale species, locations, and season using VHR imagery between 2020 and March 2025.

Species	Average Length	Location	Whale Watching Period	Citation
Beluga whale	3–5 m	Churchill River estuary, Manitoba, Canada	July	[330]
		Clearwater Fiord, Baffin Island, Nunavut, Canada	August	[325]
		Cumberland Sound, Nunavut, Canada	July–September	[95], [93], [324], [99], [326]
		Somerset Island estuaries, Nunavut, Canada	August–September	[322]
		Cook Inlet, Alaska, United States	June–August	[44]
		Baydaratskaya Inlet and Yugor Peninsula, Kara Sea, Russia	July	[329]
Blue whale	24–26 m	Monterey and Santa Barbara, California, United States	August–September	[92]
Fin whale	20–22 m	Indian Ocean (~27°16′S, 79°36′E)	March–May	[97]
		Pelagos Sanctuary, Ligurian Sea	June	[100]
Gray whale	13–14 m	Baja California lagoons, Mexico	January–February	[91], [11], [100]
Humpback whale	14–15 m	Coral Sea, Australia	September–November	[11]
		James Price Point and Camden Sound, Western Australia	July–October	[327]
		North Stradbroke Island, Queensland, Australia	June–July	[144]
		Cape Race, Newfoundland, Canada	July–August	[144]
		Gulf of Saint Lawrence, Canada	August	[94]
		Gerlache Strait, Western Antarctic Peninsula	February–March	[98], [92]
		Indian Ocean (~27°16′S, 79°36′E)	January	[96]
		Maui, Hawaii, United States	January–December	[89], [100], [11]
		Poipu Shores and Princeville, Kauai, Hawaii, United States	January–February	[144]
		Santa Barbara, California, United States	August–September	[92]
Minke whale	7–8 m	Western Antarctic Peninsula	March	[92]
Narwhal	4–5 m	Tremblay Sound, Baffin Island, Nunavut, Canada	August	[325]
North Atlantic right whale	13–16 m	Cape Cod Bay, Massachusetts, United States	February–April	[44]
Sei whale	14–16 m	Gulf of Penas, Chilean Patagonia	March–April	[104]
Southern right whale	14–15 m	Golfo Nuevo, Península Valdés, Argentina	May–December	[88], [89], [100], [11]
		Agulhas ecoregion, South Africa	June–December	[332]
		Witsand, South Africa	July–October	[11], [100]
		Enderby Island, New Zealand	July–September	[11], [90], [100]

C

Appendix C - Software Architecture Diagrams

This appendix provides the software architecture diagrams referred to in subsection 5.1.2. It complements the methodological description by visualizing the main components of the integrated framework, including the orbital simulation, the off-nadir imaging pipeline, and the separate cross-validation pipeline used for detector development. Where relevant, the appendix also includes pseudo-code to clarify the implemented software logic and data flow.

C.1. Pseudo-code

Algorithm 1 Tip-and-Cue simulation pipeline

```
Initialize satellite orbits and whale targets
for each simulation time step do
    Propagate satellite states
    Propagate whale target states
    for each Tip satellite do
        Determine targets within field of view
        Evaluate detections using predefined TPR and TNR
        if target is positively detected then
            Apply confirmation delay
            Assign and task a Cue satellite
        end if
    end for
    for each Cue satellite do
        Select next feasible task
        Compute required Euler angles for pointing
        Perform pointing and stabilization
        if target enters field of view then
            Generate off-nadir image
            Evaluate onboard detection
            Log observation and confirmation
        end if
    end for
end for
Aggregate mission-level metrics
```

C.2. Orbital Simulation Framework

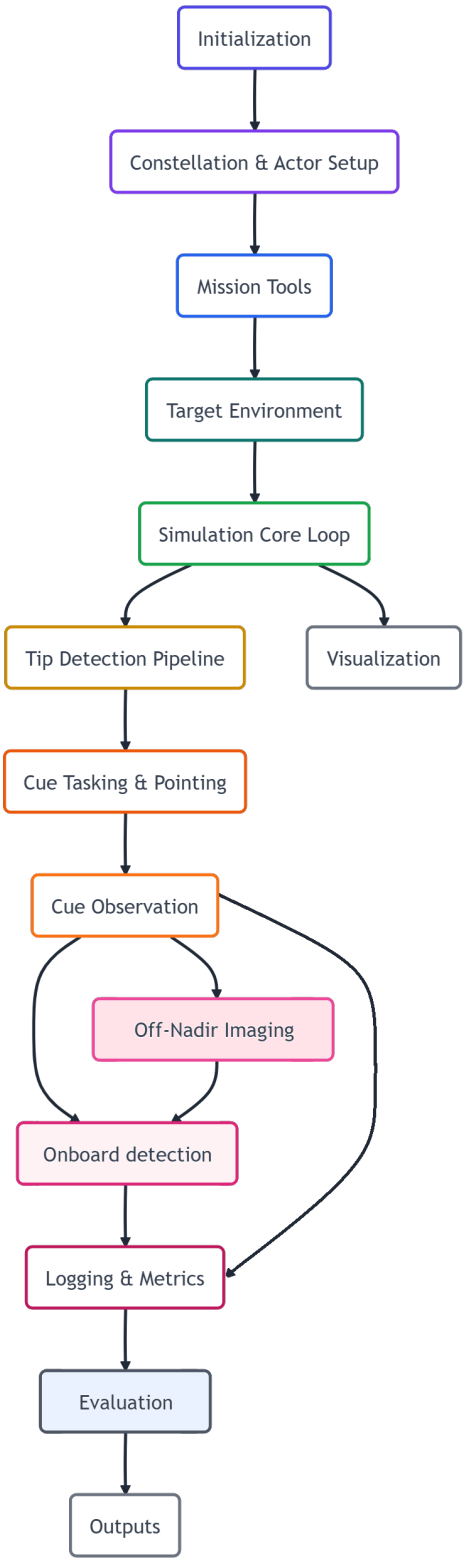
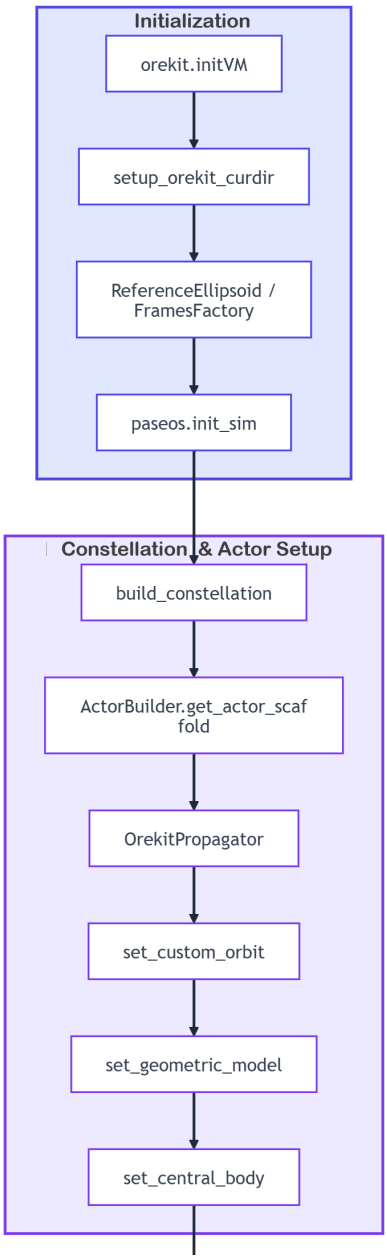
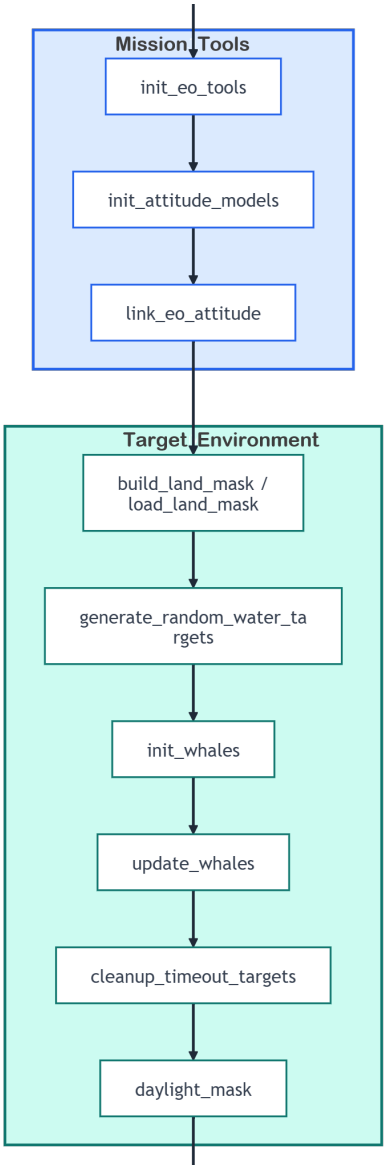
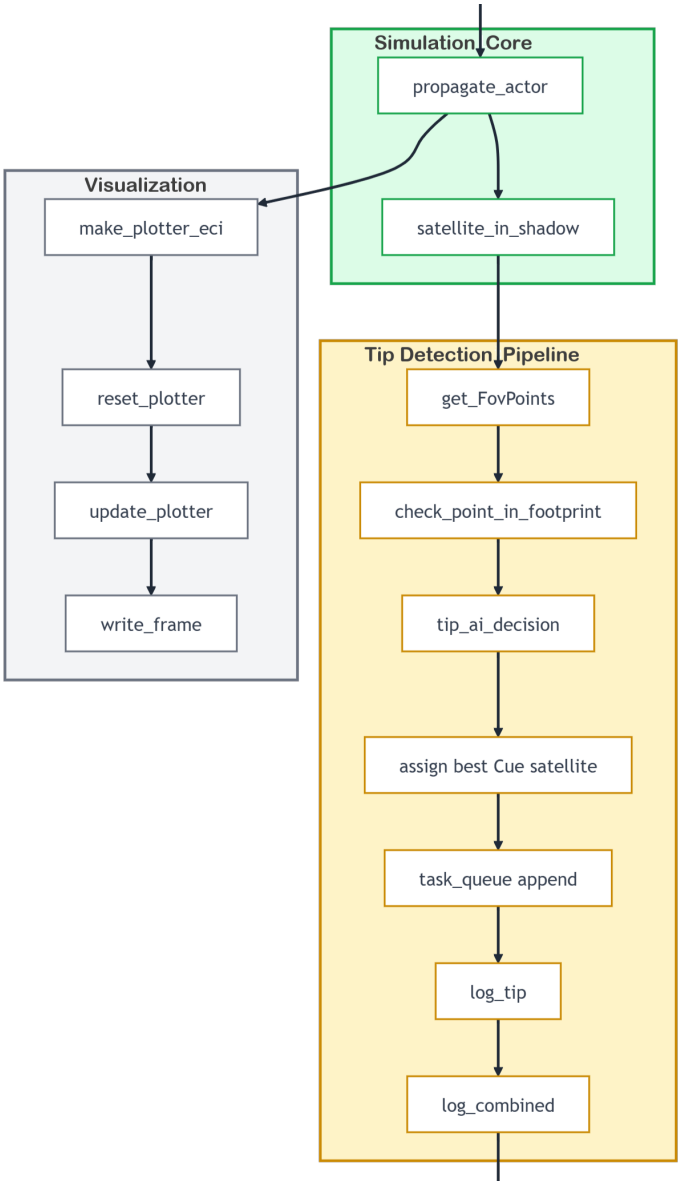
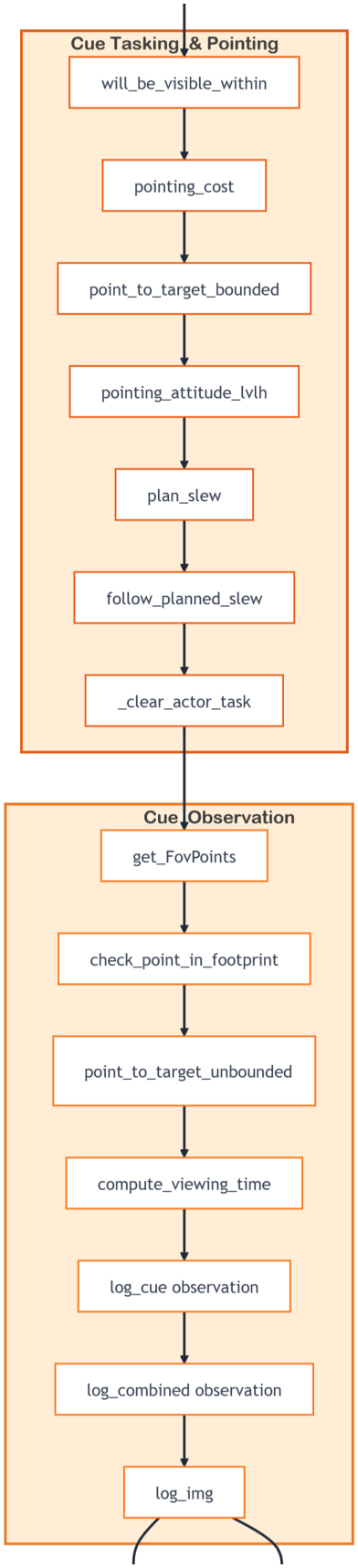


Figure C.1: High-Level Software Architecture Diagram for the Orbital Simulation framework. This diagram outlines the entire mission simulation, including satellite propagation, tasking, and target detection.


Initialization
orekit.initVM
setup_orekit_curdir
ReferenceEllipsoid / FramesFactory
paseos.init_sim
Constellation & Actor Setup
build_constellation
ActorBuilder.get_actor_scaffold
OrekitPropagator
set_custom_orbit
set_geometric_model
set_central_body


Mission_Tools
init_eo_tools
init_attitude_models
link_eo_attitude
Target_Environment
build_land_mask / load_land_mask
generate_random_water_targets
init_whales
update_whales
cleanup_timeout_targets
daylight_mask


Simulation Core
propagate_actor
satellite_in_shadow
Visualization
make_plotter_eci
reset_plotter
update_plotter
write_frame
Tip Detection Pipeline
get_FovPoints
check_point_in_footprint
tip_ai_decision
assign best Cue satellite
task_queue append
log_tip
log_combined


Cue Tasking & Pointing
will_be_visible_within
pointing_cost
point_to_target_bounded
pointing_attitude_lvlh
plan_slew
follow_planned_slew
_clear_actor_task
Cue Observation
get_FovPoints
check_point_in_footprint
point_to_target_unbounded
compute_viewing_time
log_cue observation
log_combined observation
log_img

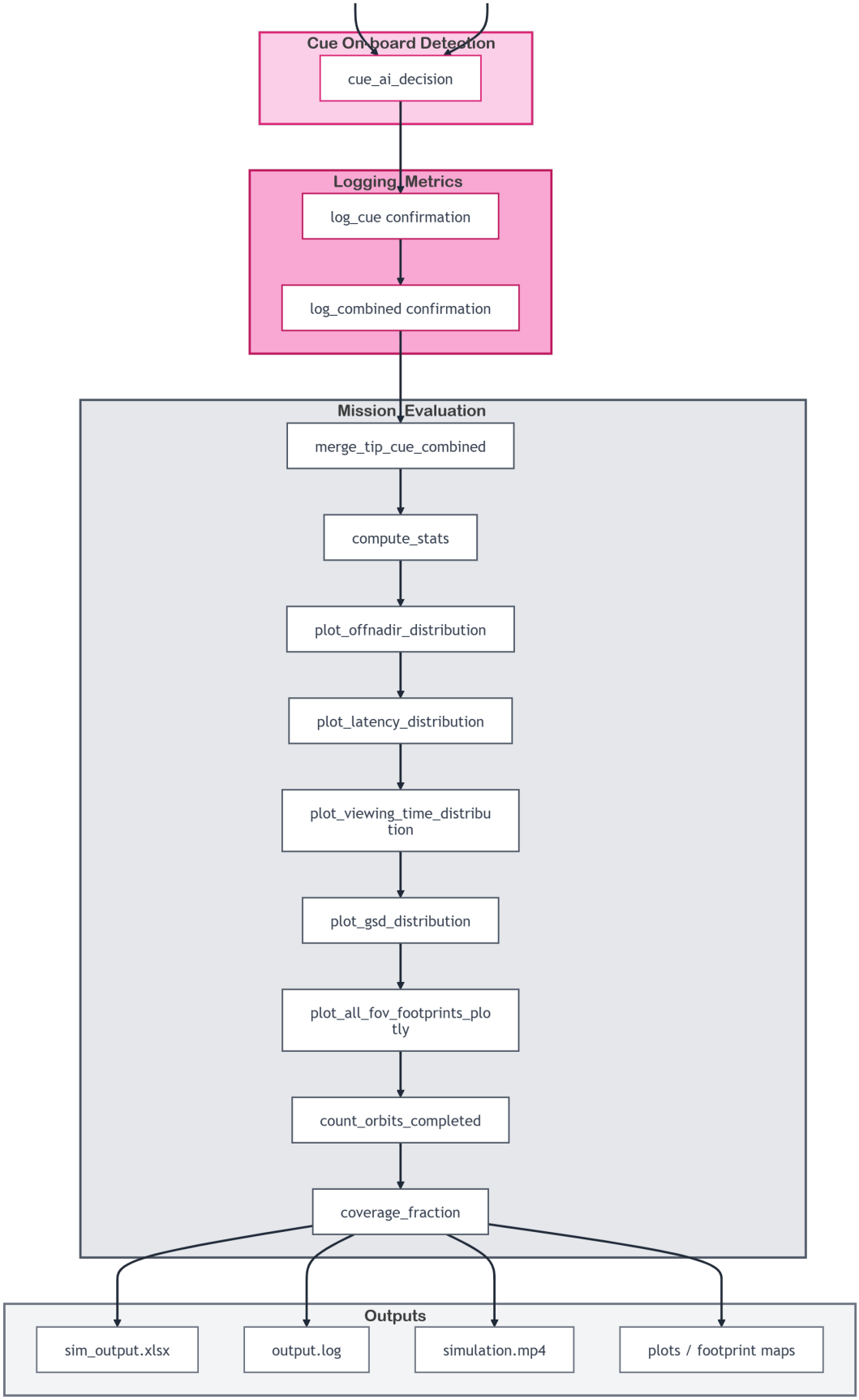


Figure C.2: Detailed Software Architecture Diagram for the Orbital Simulation framework. This diagram outlines the full software system, covering satellite propagation, mission configuration, and tasking components.

C.3. Off-Nadir Imaging Pipeline

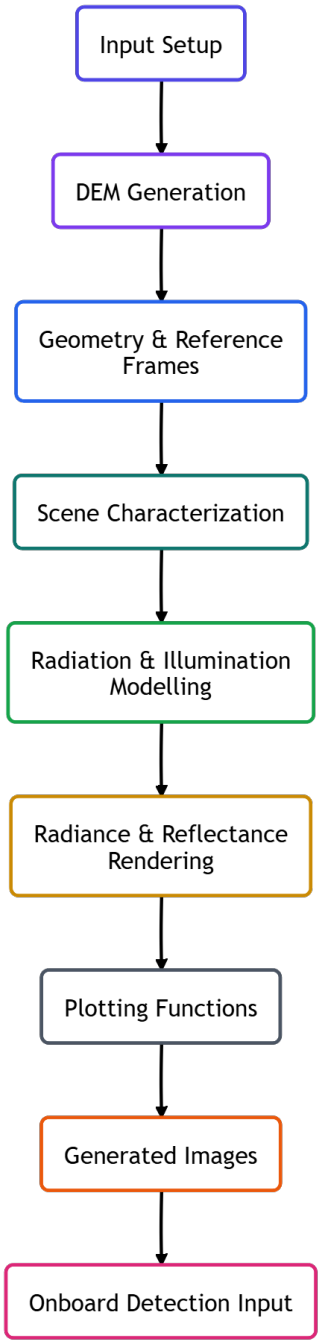
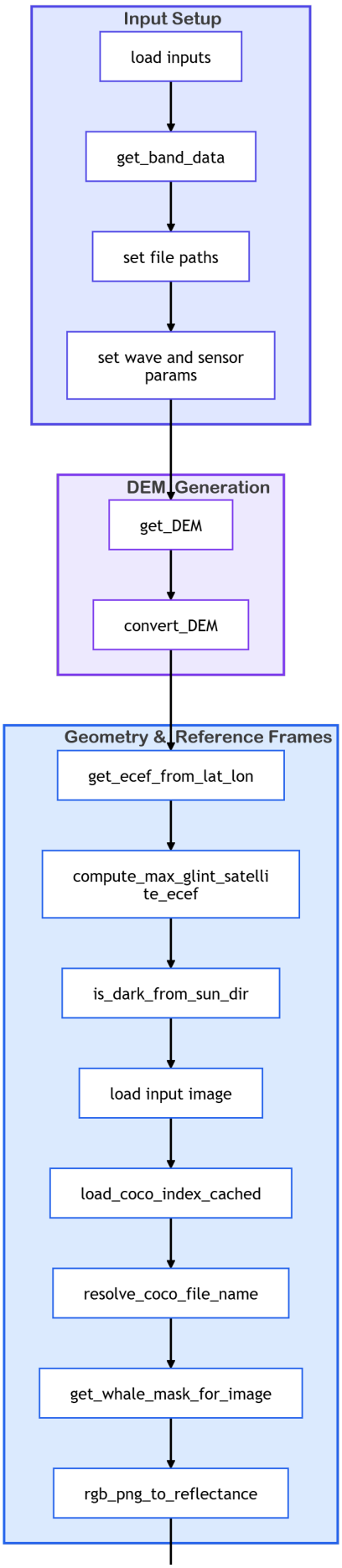
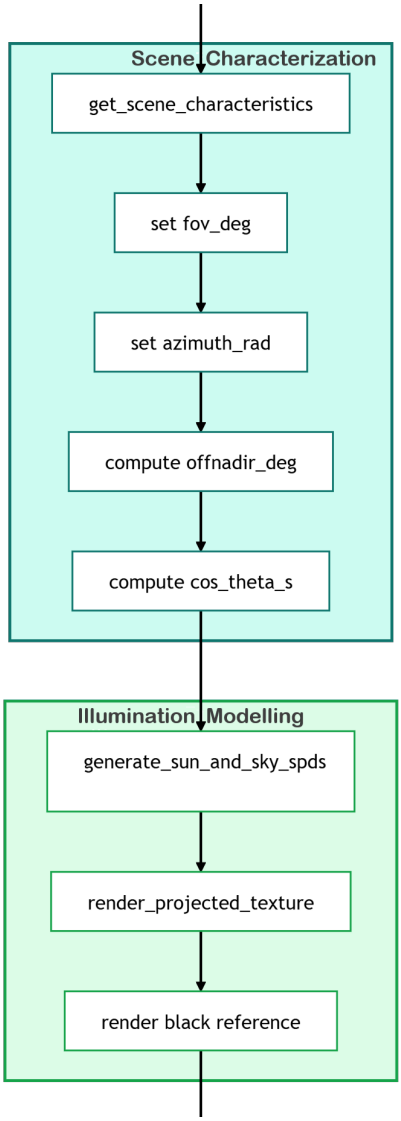


Figure C.3: High-Level Software Architecture Diagram for the Off-Nadir Image Rendering pipeline. This diagram illustrates the flow of image generation based on satellite geometry and off-nadir angle, which affects the image quality and coverage.


Input Setup
load inputs
get_band_data
set file paths
set wave and sensor params
DEM Generation
get_DEM
convert_DEM
Geometry & Reference Frames
get_ecef_from_lat_lon
compute_max_glint_satellite_ecef
is_dark_from_sun_dir
load input image
load_coco_index_cached
resolve_coco_file_name
get_whale_mask_for_image
rgb_png_to_reflectance


Scene_Characterization
get_scene_characteristics
set fov_deg
set azimuth_rad
compute offnadir_deg
compute cos_theta_s
Illumination_Modelling
generate_sun_and_sky_spds
render_projected_texture
render black reference

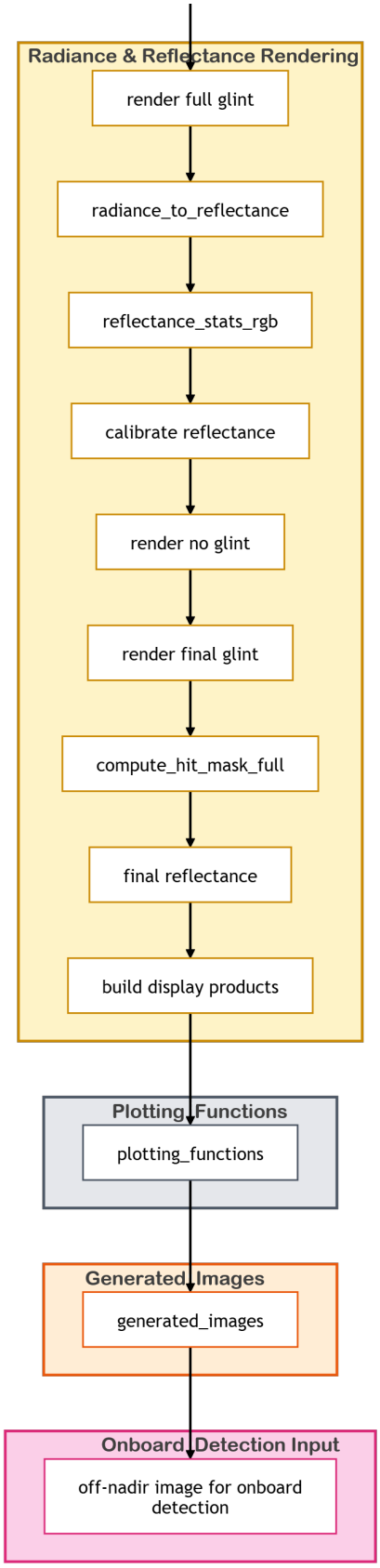


Figure C.4: Detailed Software Architecture Diagram for the Off-Nadir Image Rendering pipeline. This diagram details the image generation process, including radiative effects and geometry corrections.

C.4. Cross-Validation Pipeline

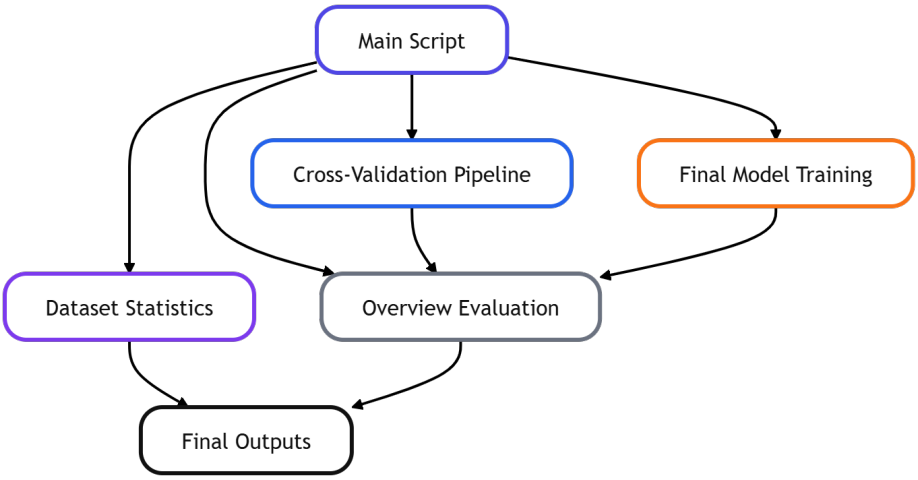


Figure C.5: High-Level Software Architecture Diagram for the Cross-Validation pipeline. This diagram shows the components responsible for training, validating, and evaluating the models across multiple data folds.

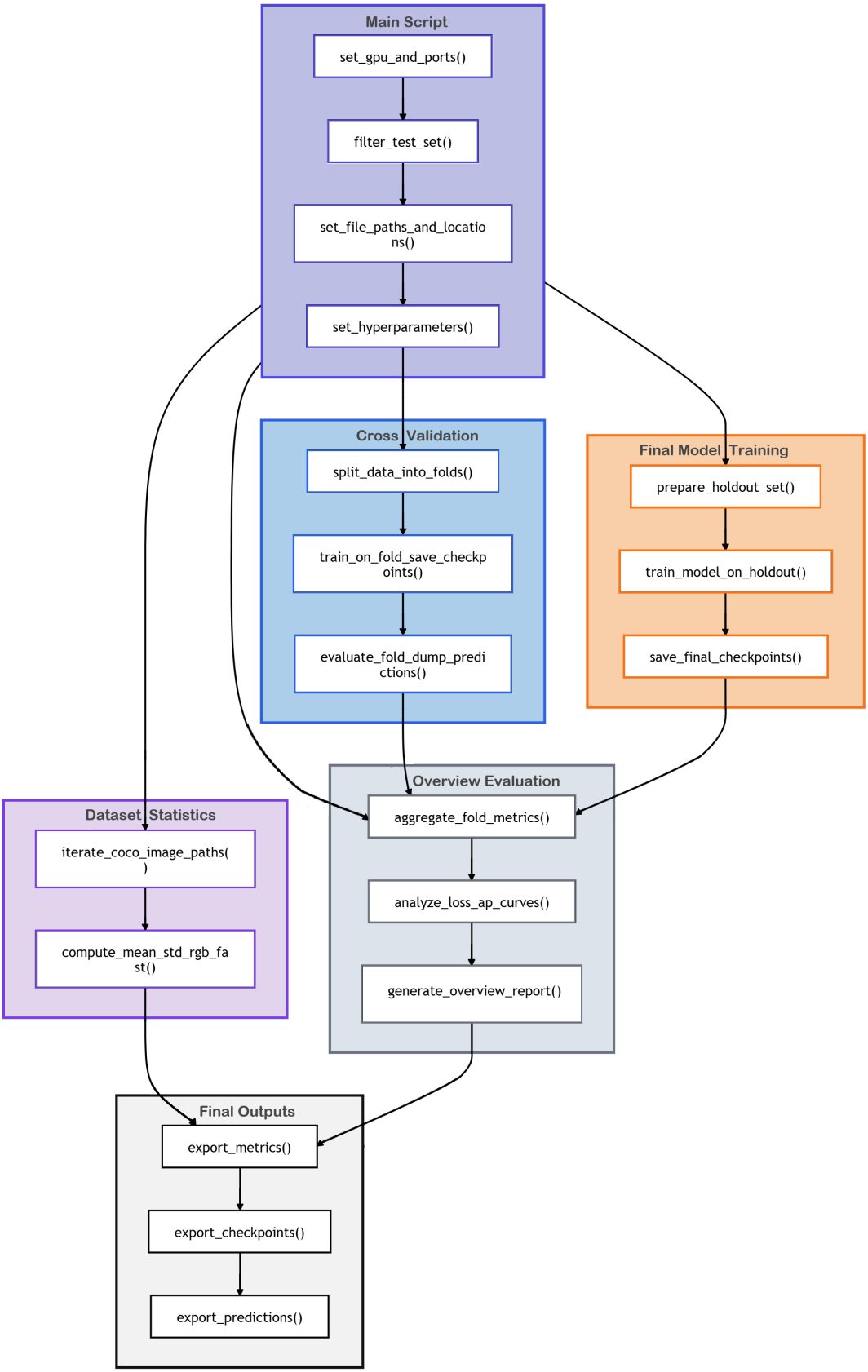


Figure C.6: Detailed Software Architecture Diagram for the Cross-Validation pipeline. This diagram illustrates the detailed flow for model training, cross-validation, and final evaluation.

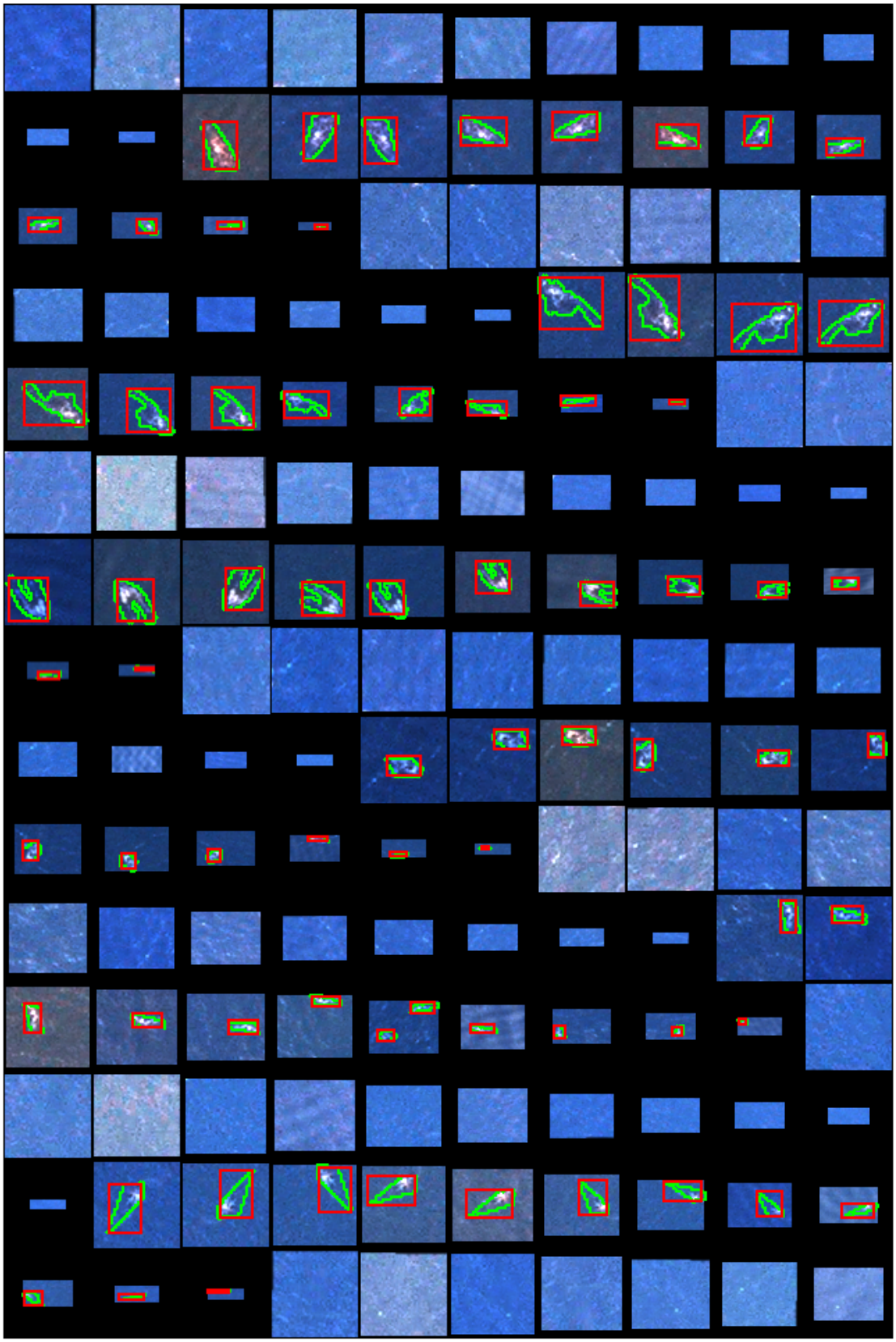
D

Appendix D - Dataset

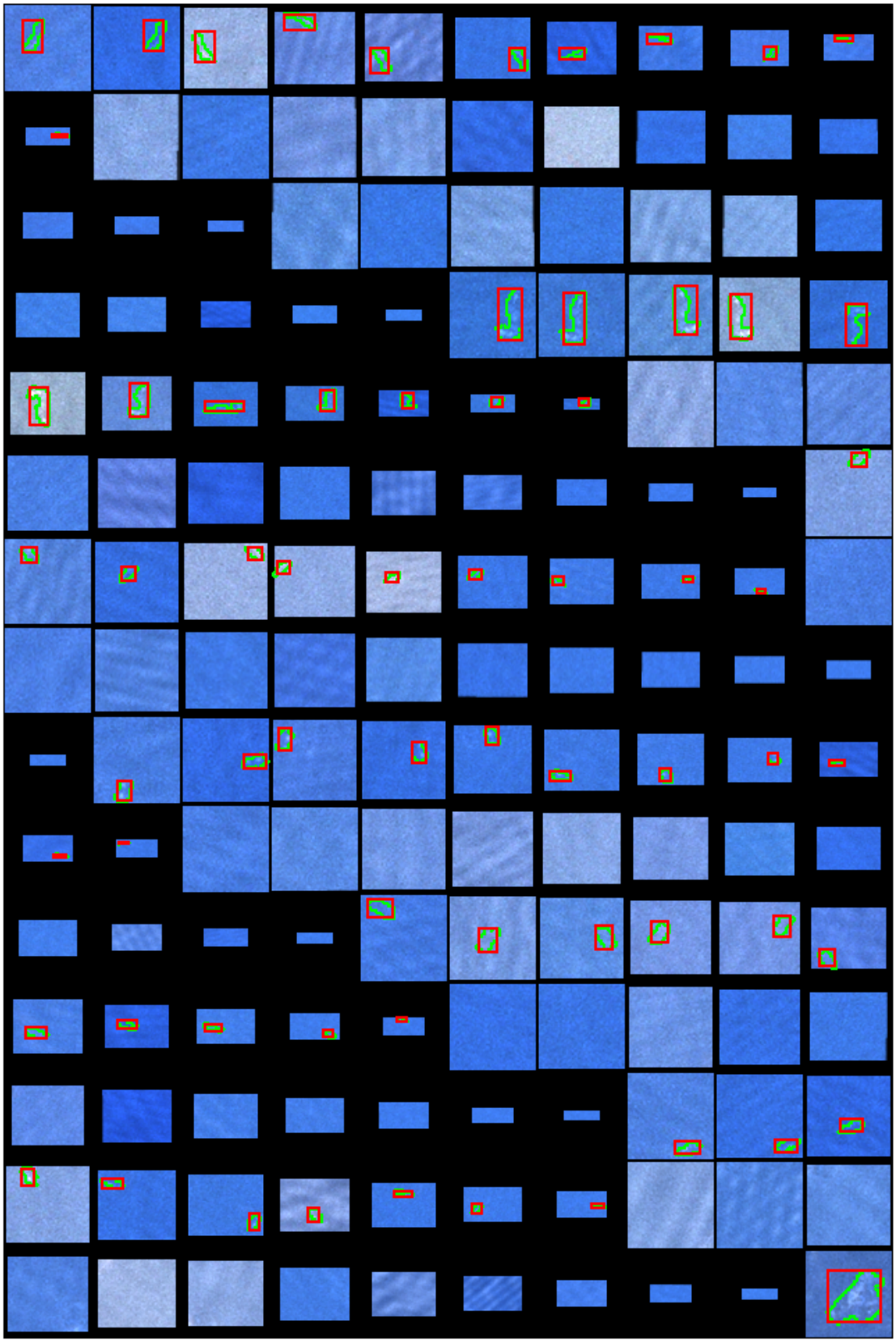
This appendix provides representative samples of the generated whale-detection dataset used throughout the thesis. It includes the default off-nadir reflectance images with radiometric effects, both with and without annotation overlays, as well as the alternative image configurations used for the imaging-effects comparison in Chapter 7.

D.1. Reflectance, Off-Nadir, with Radiometric Effects

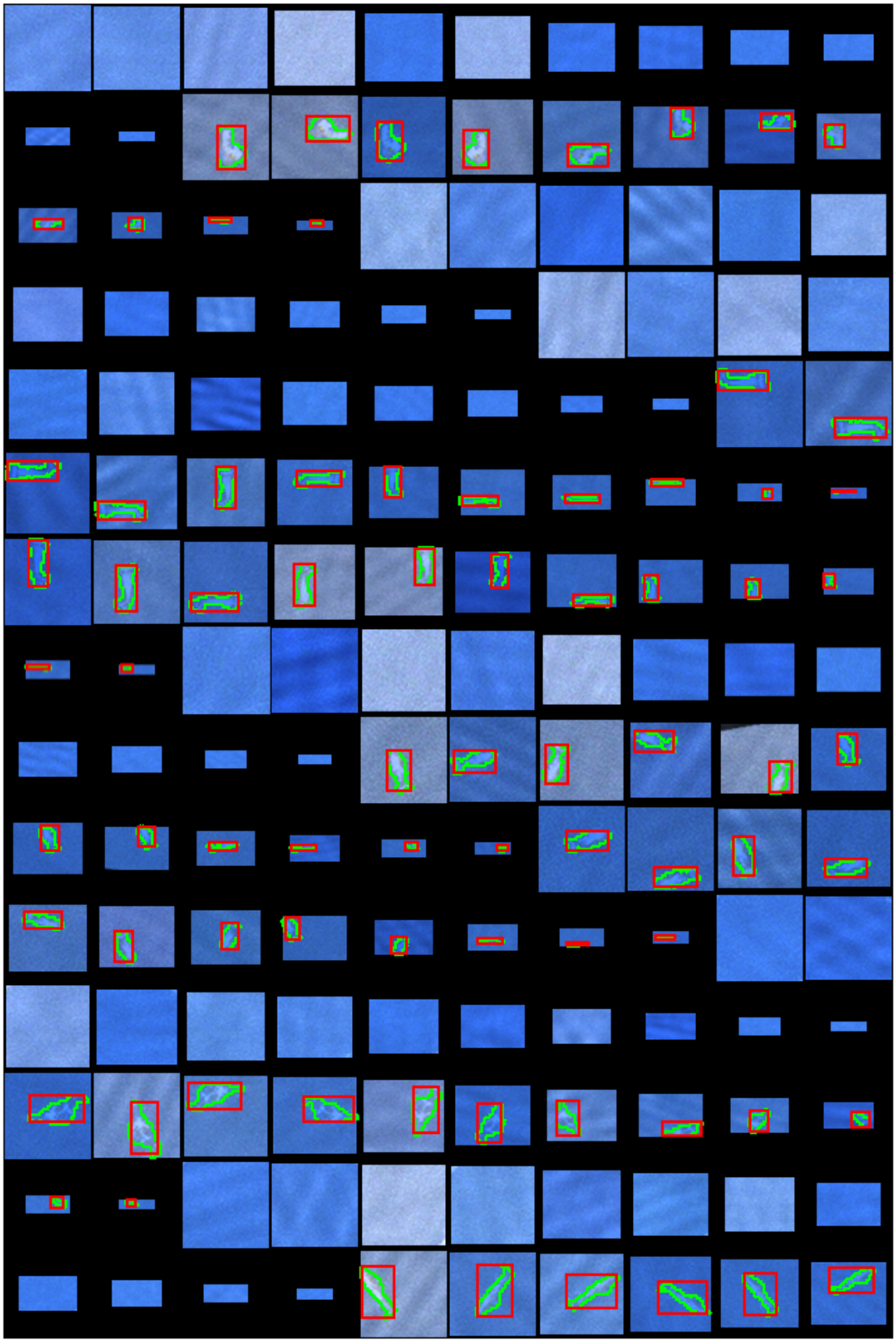
D.1.1. With Annotation



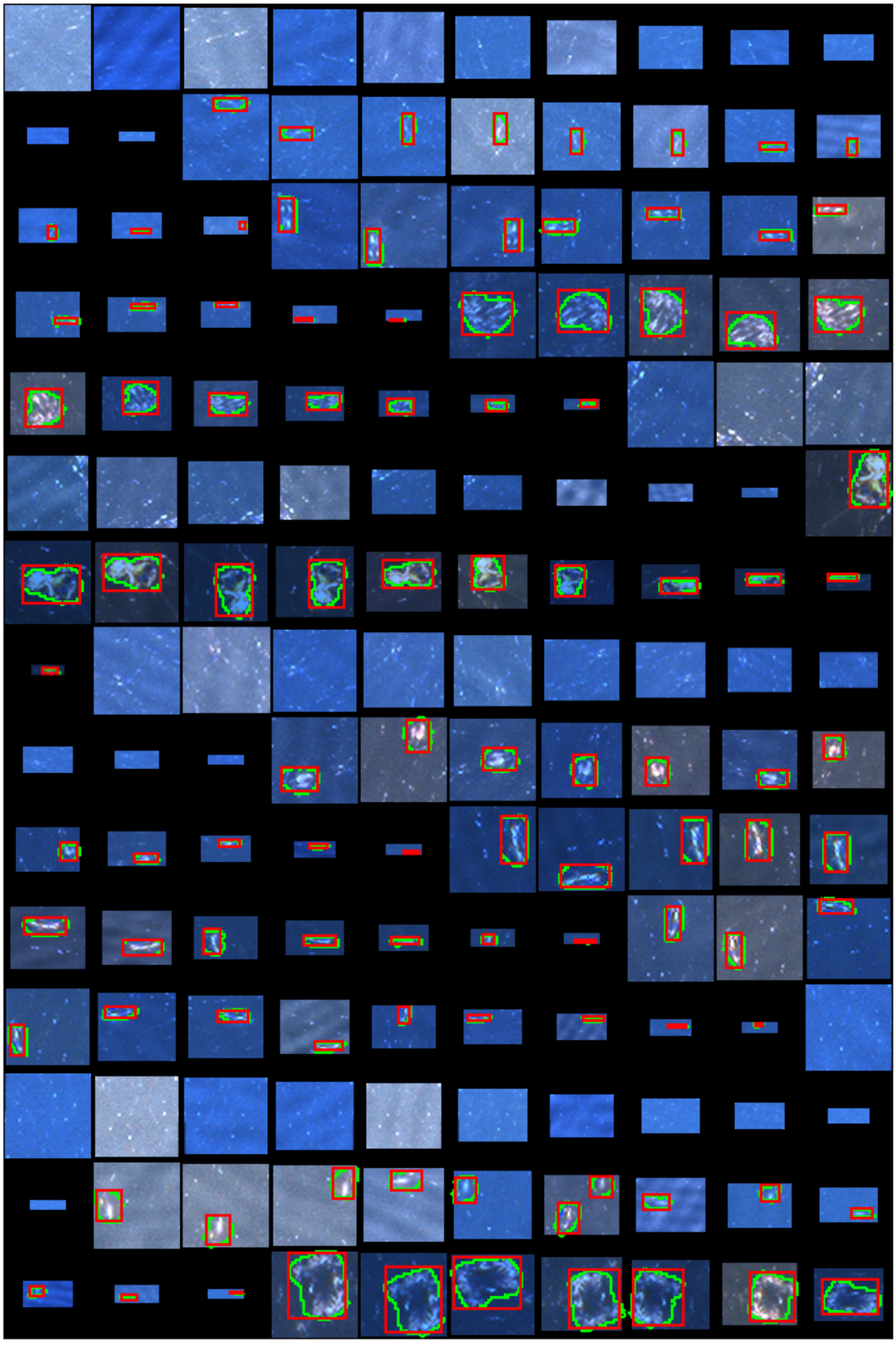
Auckland2006. *Satellite image © 2022 Maxar Technologies.*



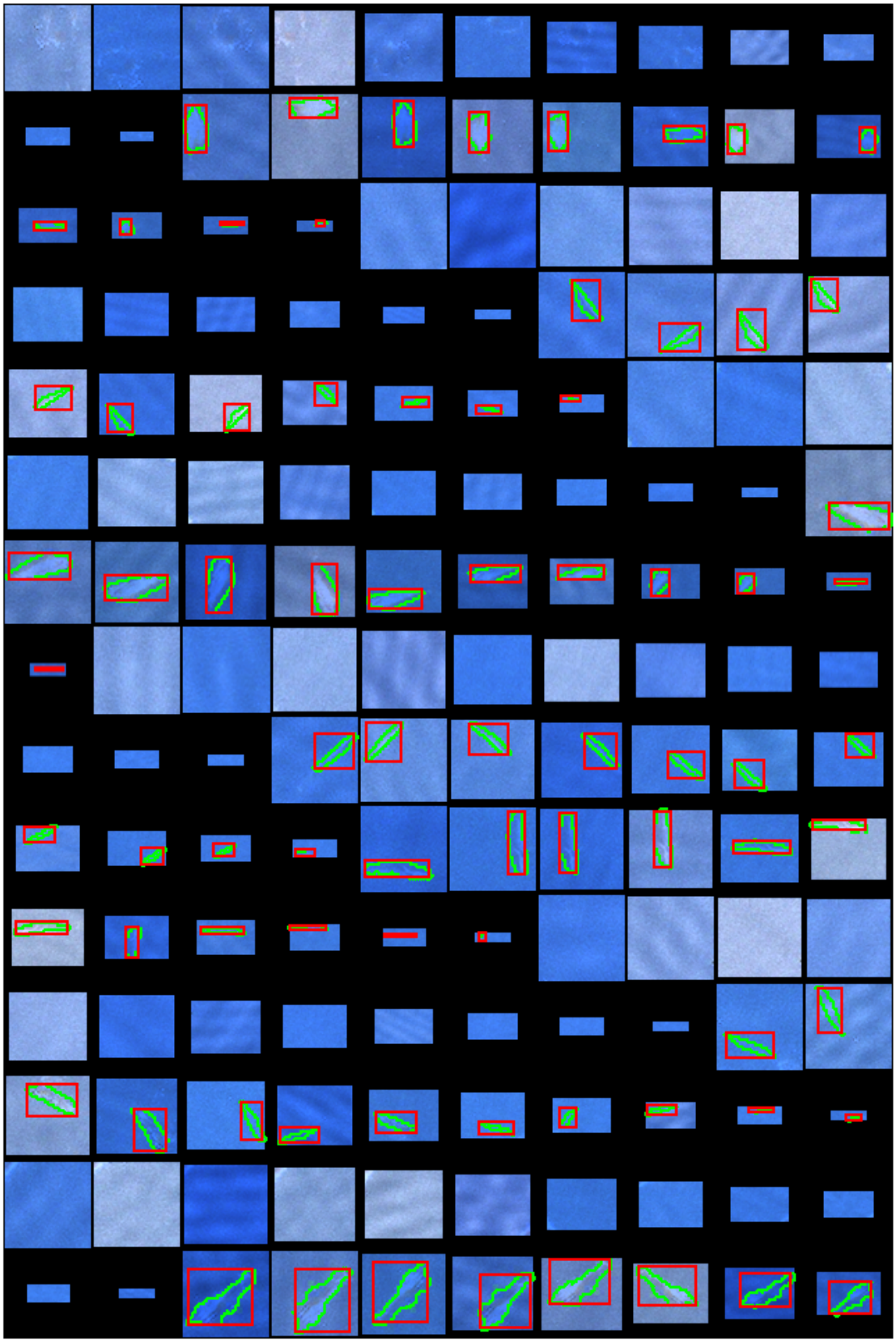
Auckland2011. *Satellite image © 2022 Maxar Technologies.*



Ignacio2017. *Satellite image © 2022 Maxar Technologies.*

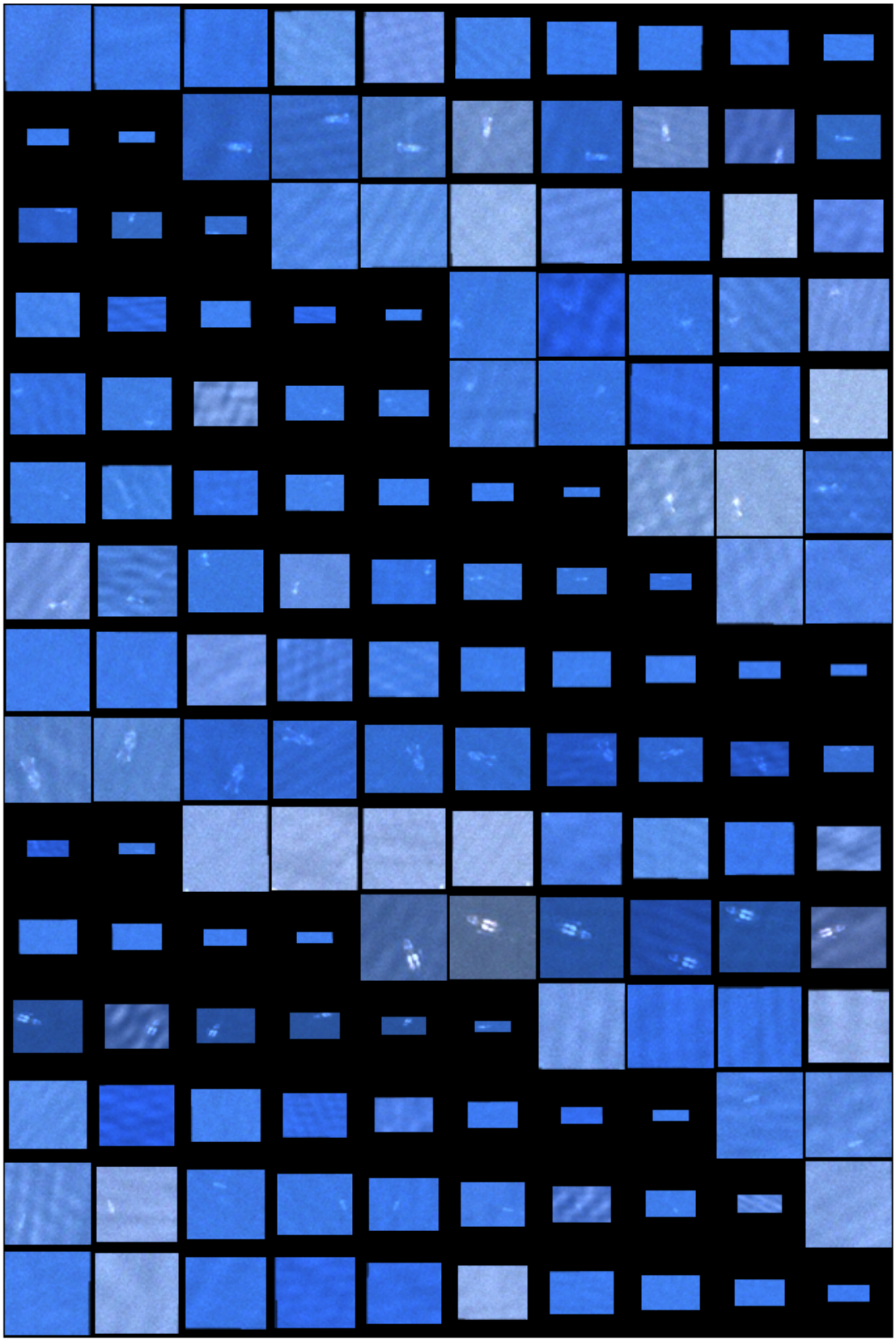


Maui2015. *Satellite image © 2022 Maxar Technologies.*

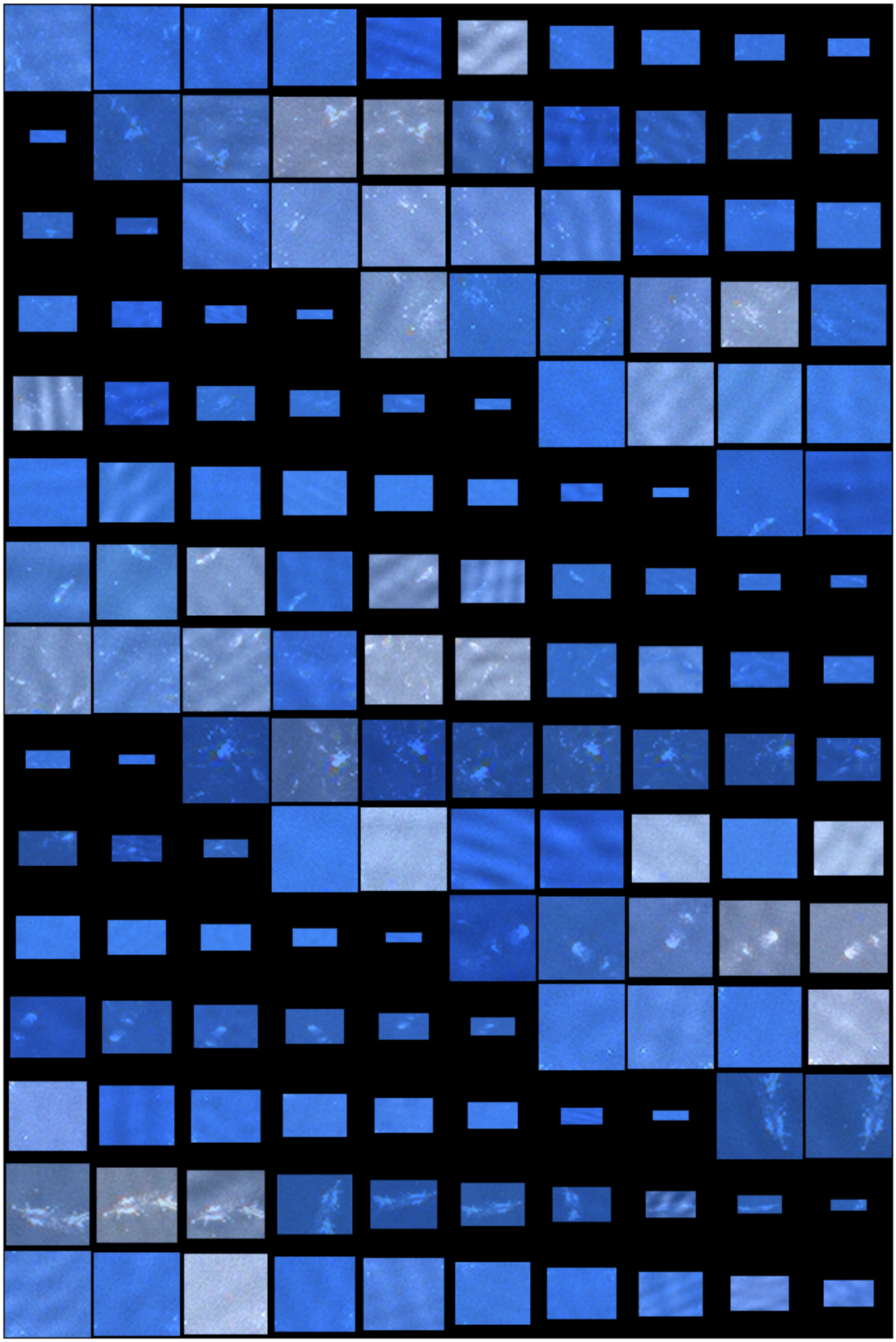


Pelagos2016. *Satellite image © 2022 Maxar Technologies.*

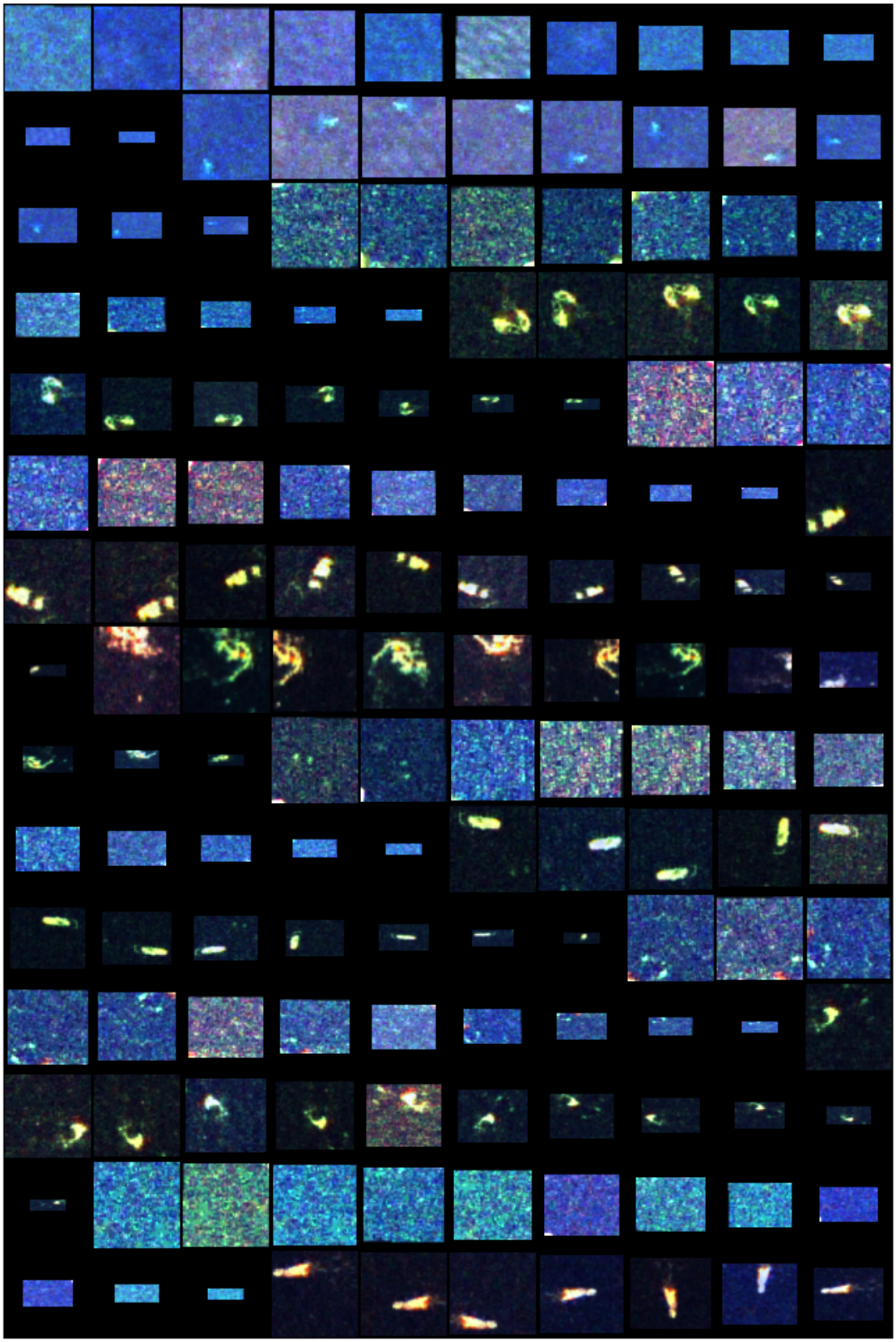
D.1.2. Without Annotations



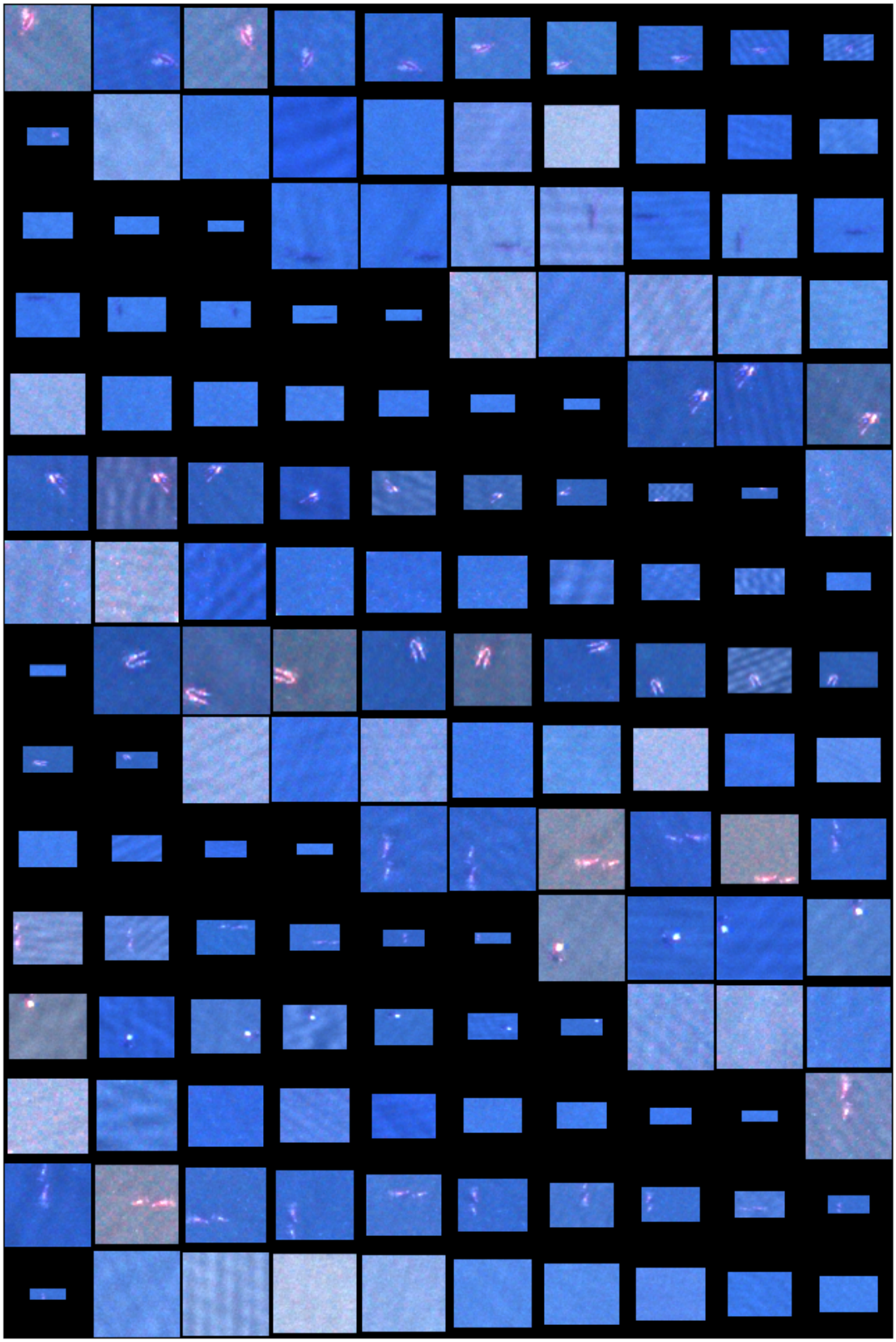
Valdes2012. *Satellite image © 2022 Maxar Technologies.*



Valdes2014. *Satellite image © 2022 Maxar Technologies.*

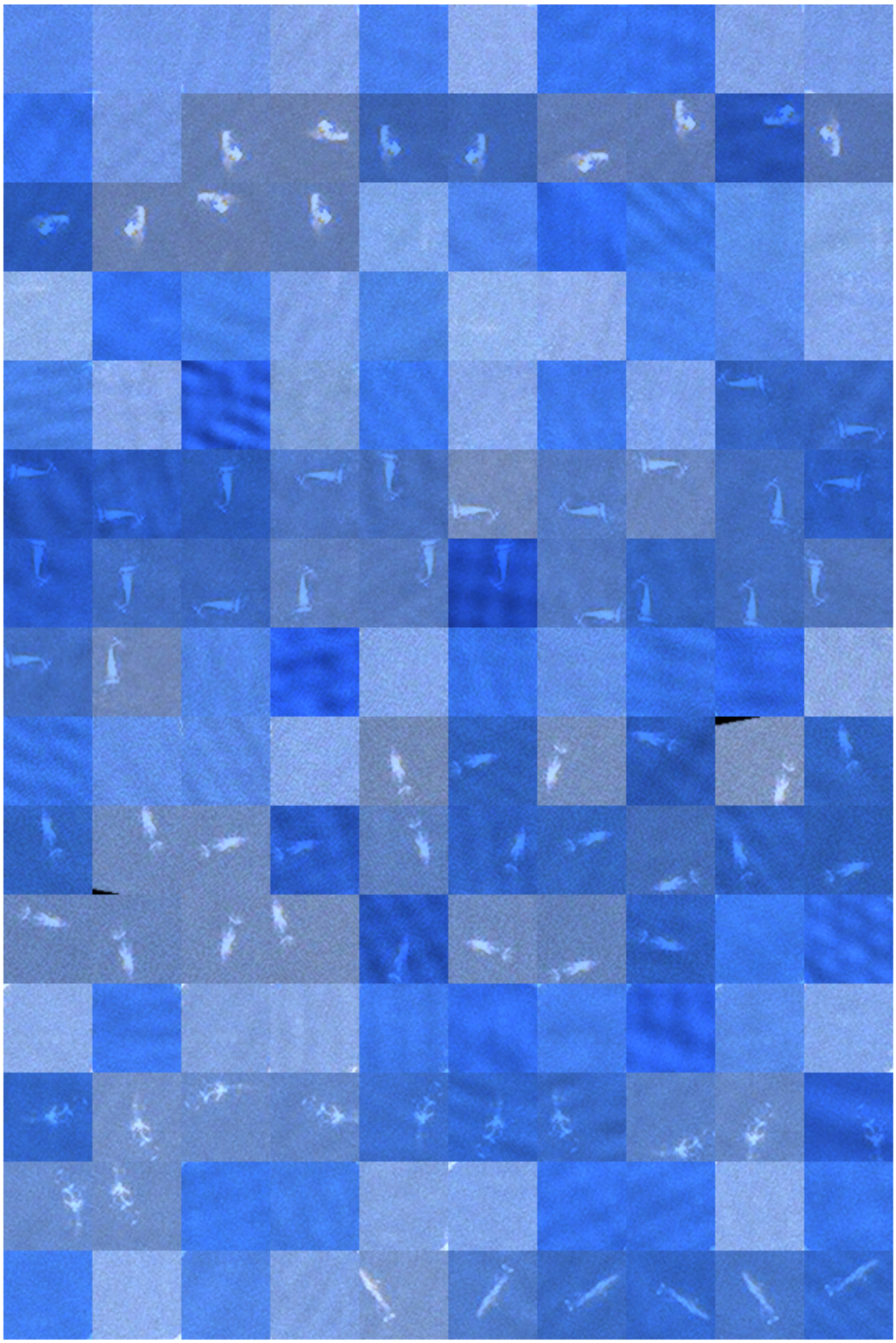


Valdes2016. *Satellite image © 2022 Maxar Technologies.*

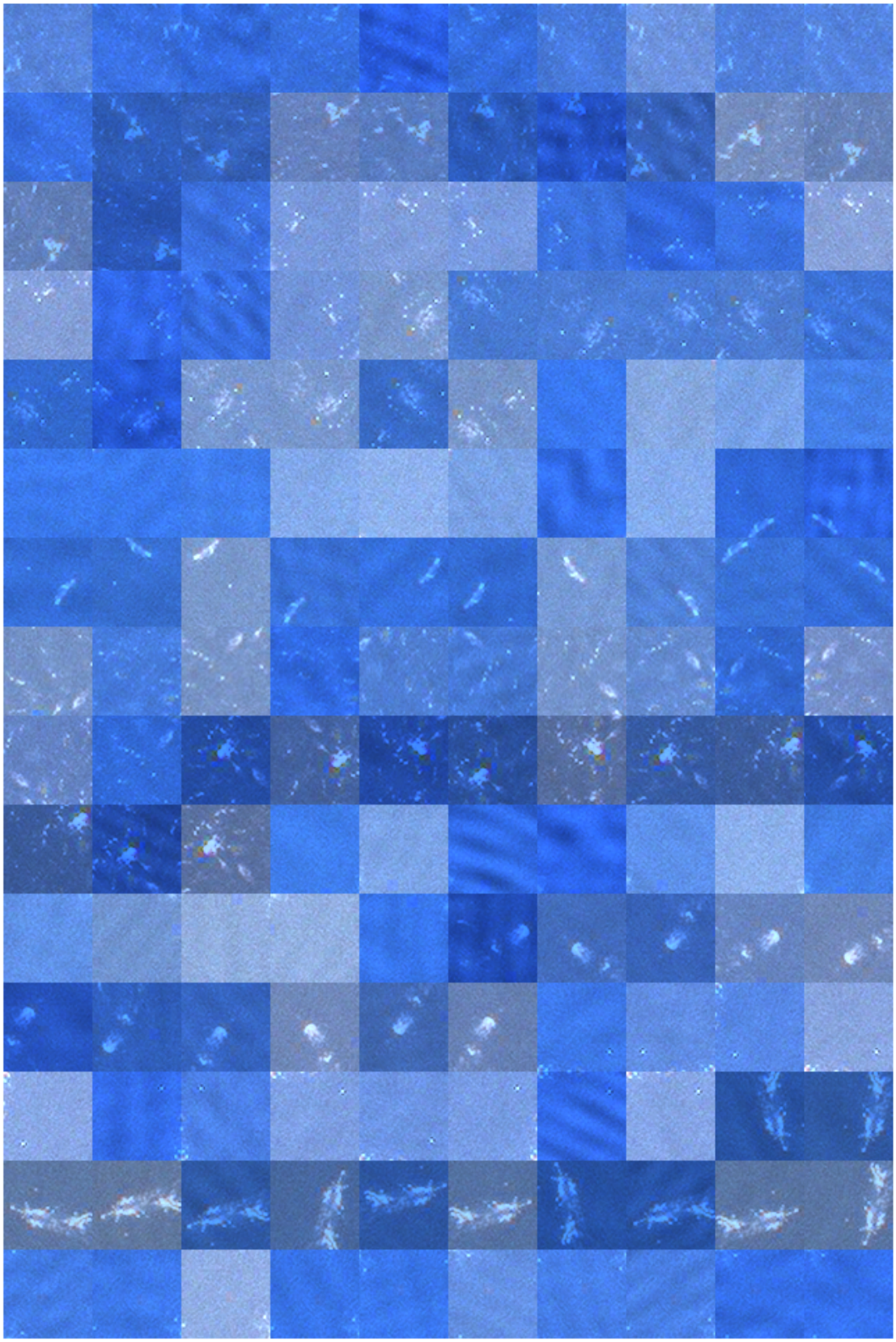


Witsand2009. *Satellite image © 2022 Maxar Technologies.*

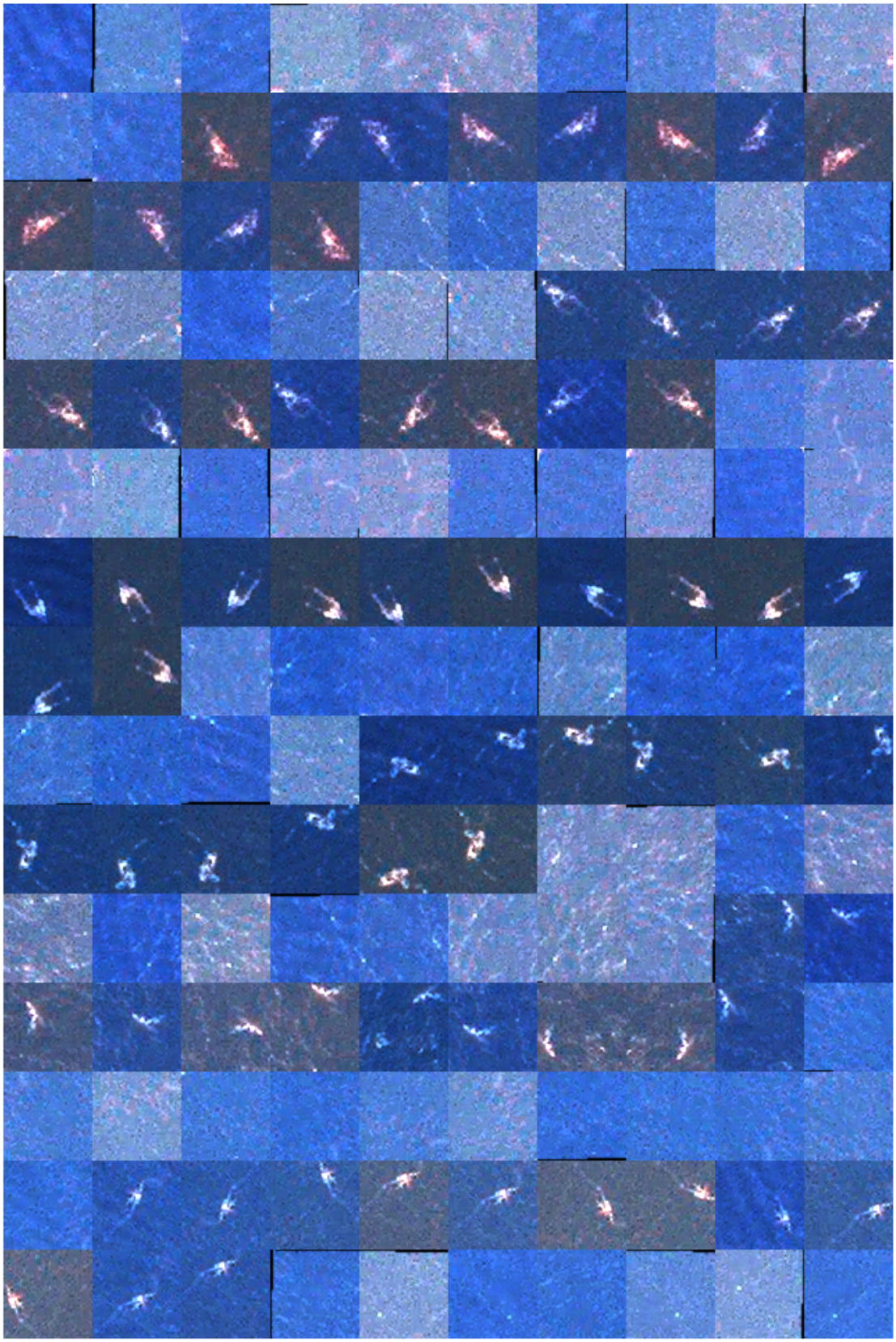
D.2. Reflectance, Nadir, with Radiometric Effects



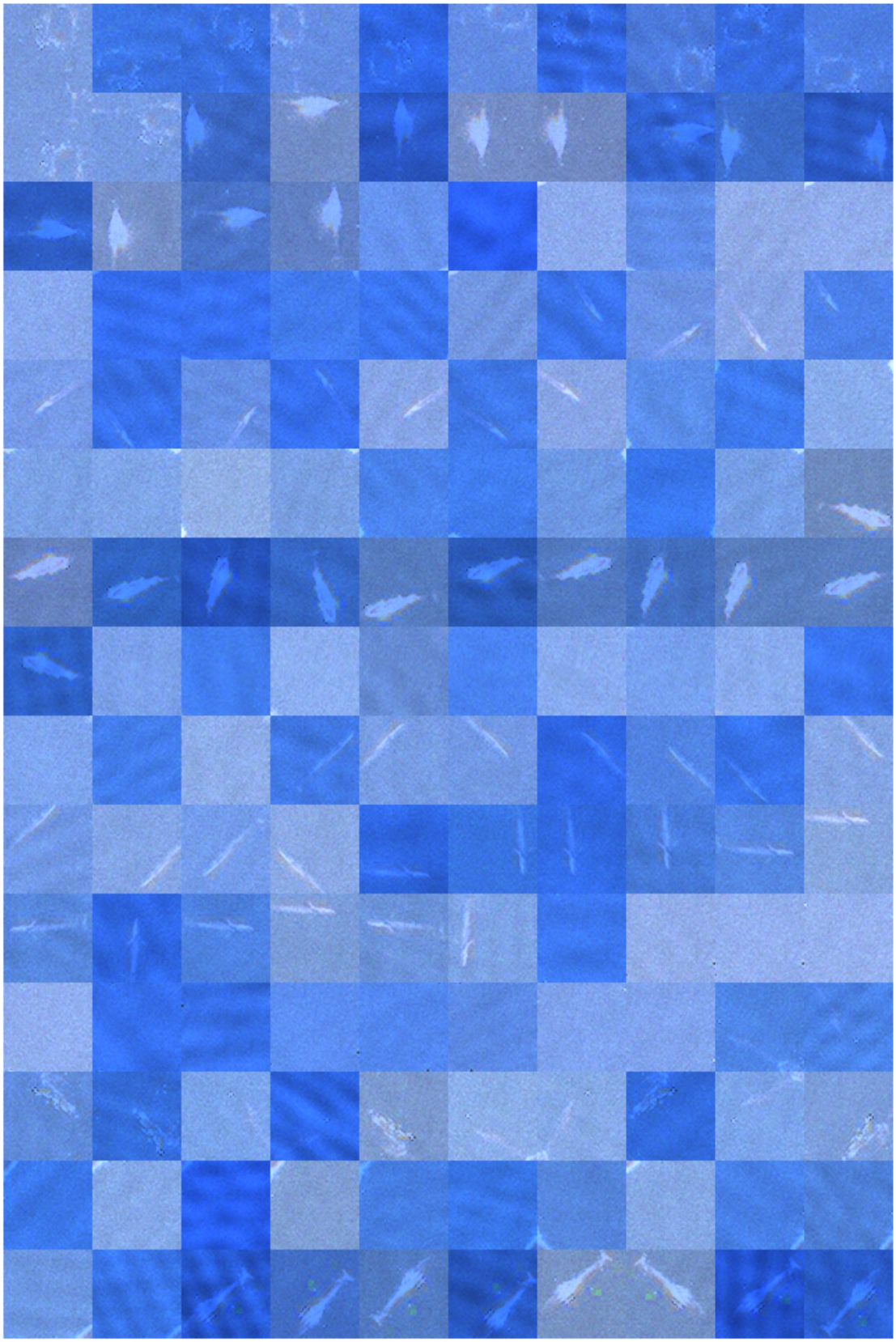
Ignacio2017. *Satellite image © 2022 Maxar Technologies.*



Valdes2014. *Satellite image © 2022 Maxar Technologies.*

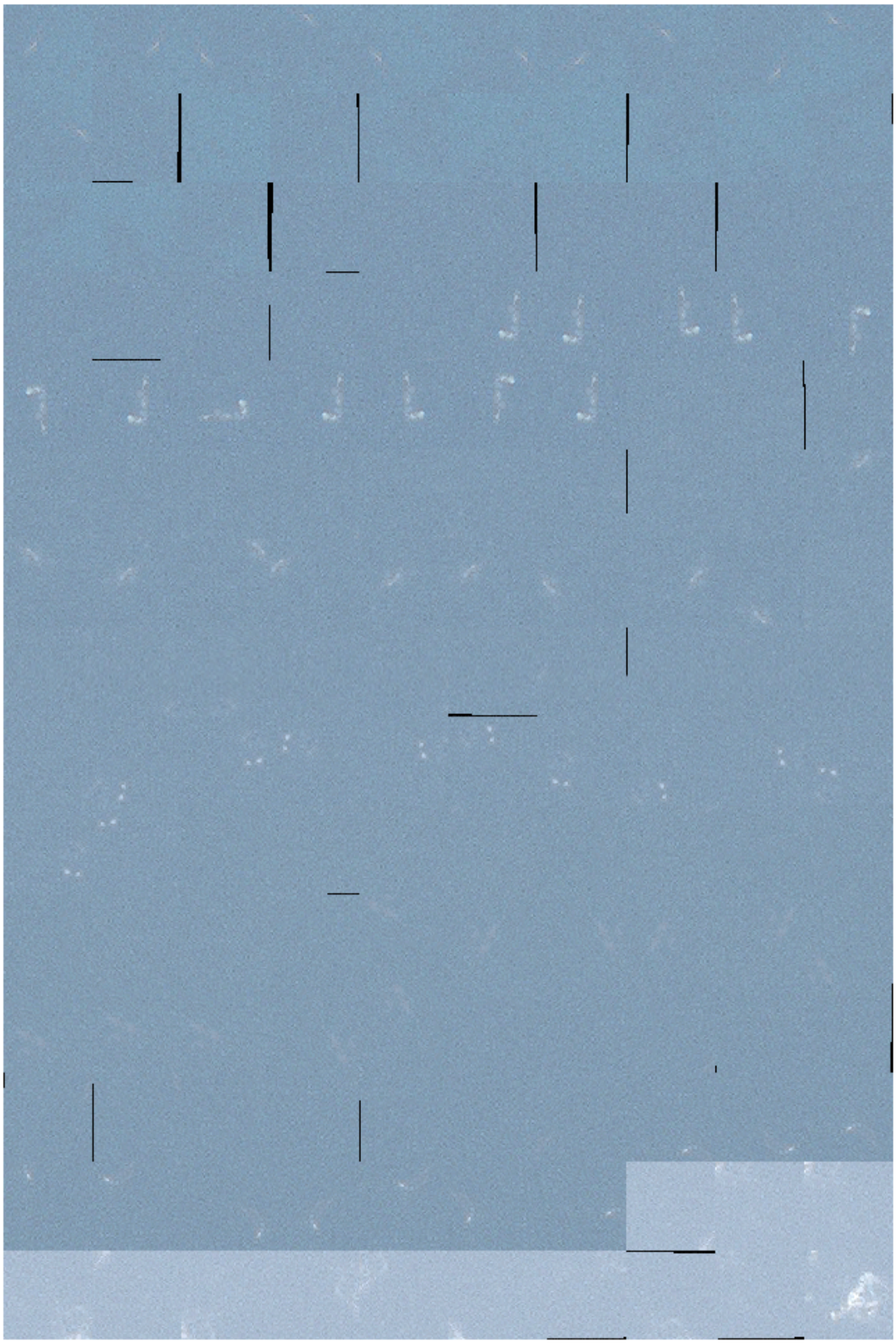


Auckland2006. *Satellite image © 2022 Maxar Technologies.*

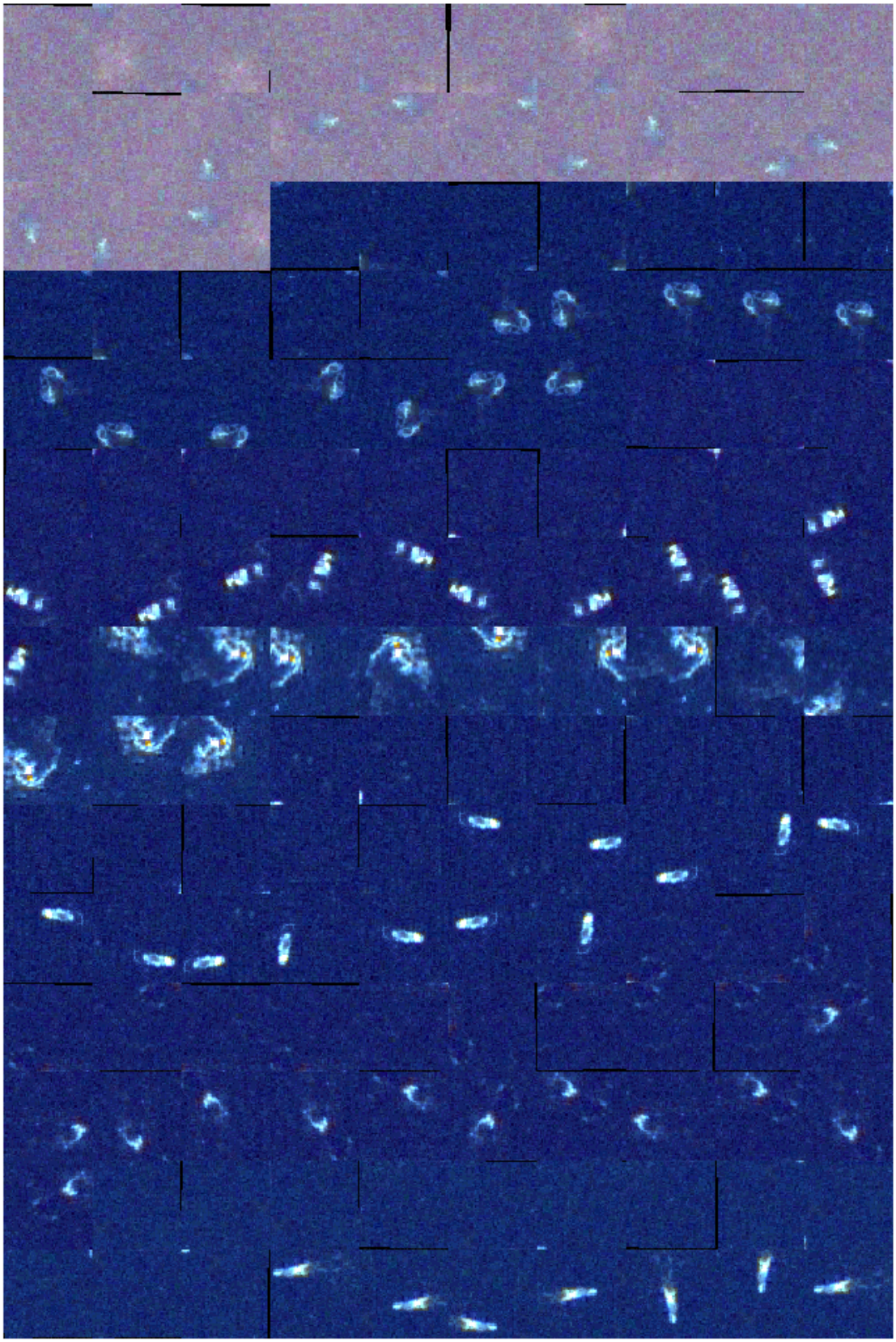


Pelagos2016. *Satellite image © 2022 Maxar Technologies.*

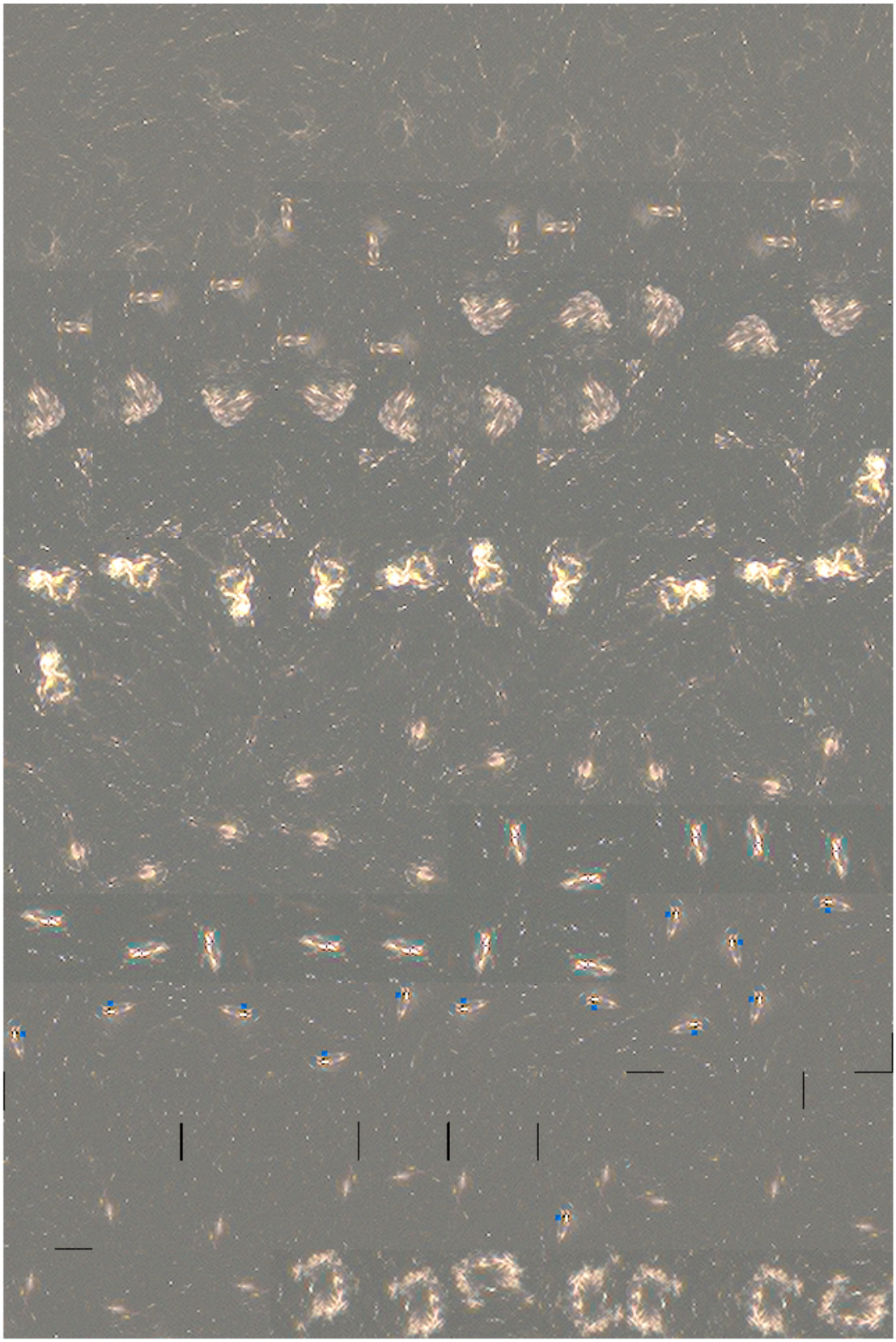
D.3. Original Input Images - Texture, Nadir, No Radiometric Effects



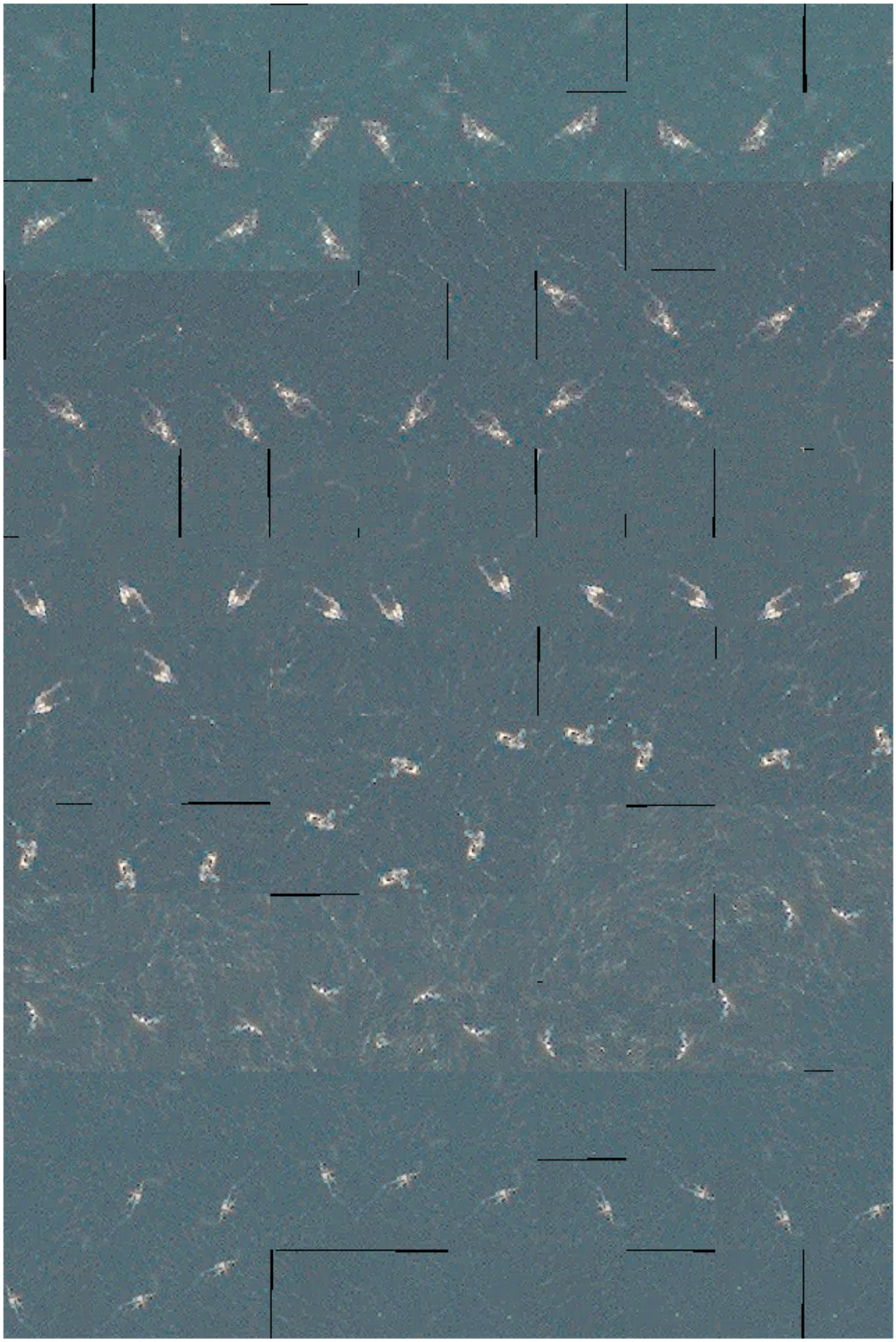
Auckland2011. *Satellite image © 2022 Maxar Technologies.*



Pelagos2016. *Satellite image © 2022 Maxar Technologies.*



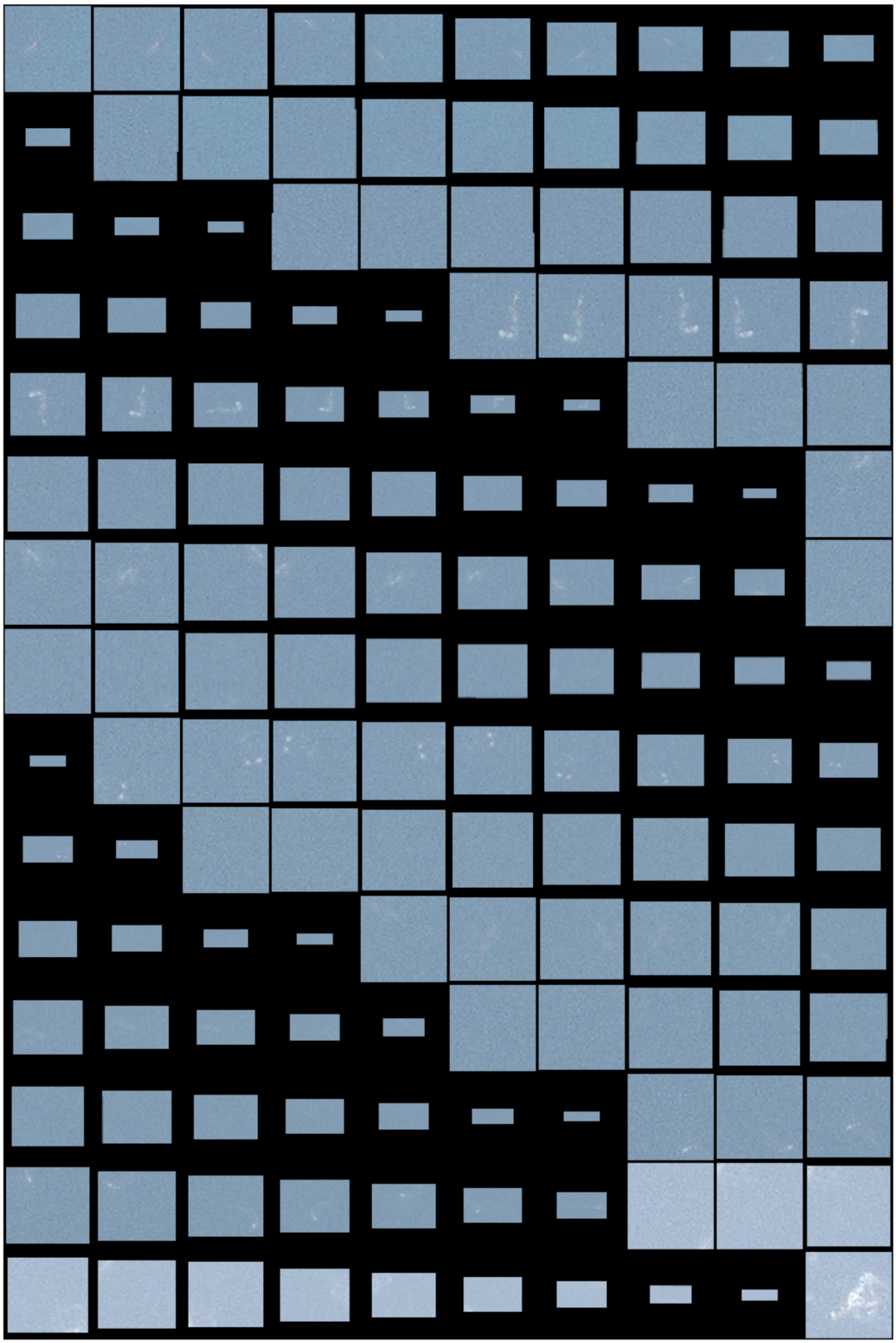
Maui2015. *Satellite image © 2022 Maxar Technologies.*



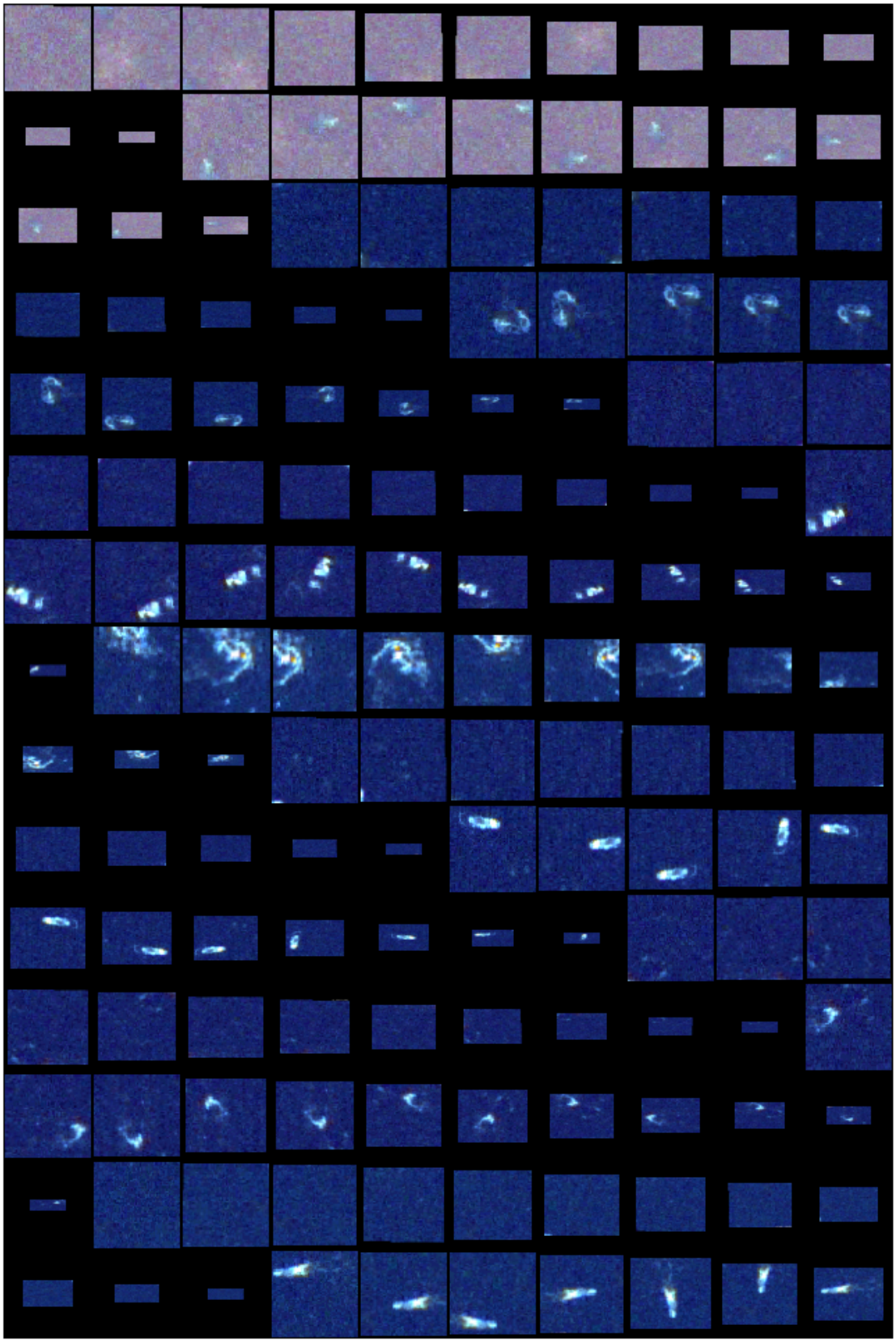
Auckland2006. *Satellite image © 2022 Maxar Technologies.*

Ignacio2017. *Satellite image © 2022 Maxar Technologies.*

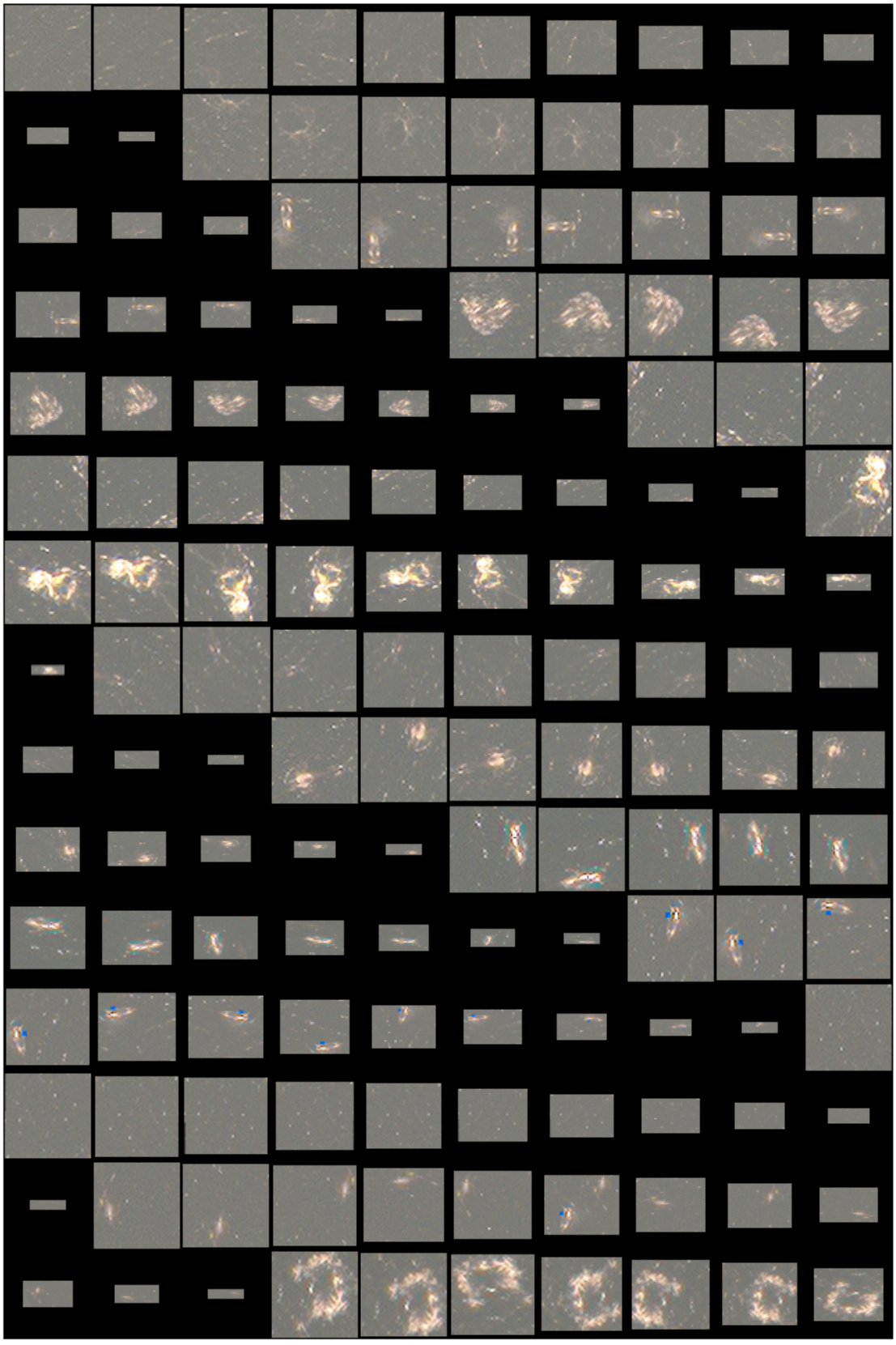
D.4. Texture, Off-Nadir, without Radiometric Effects



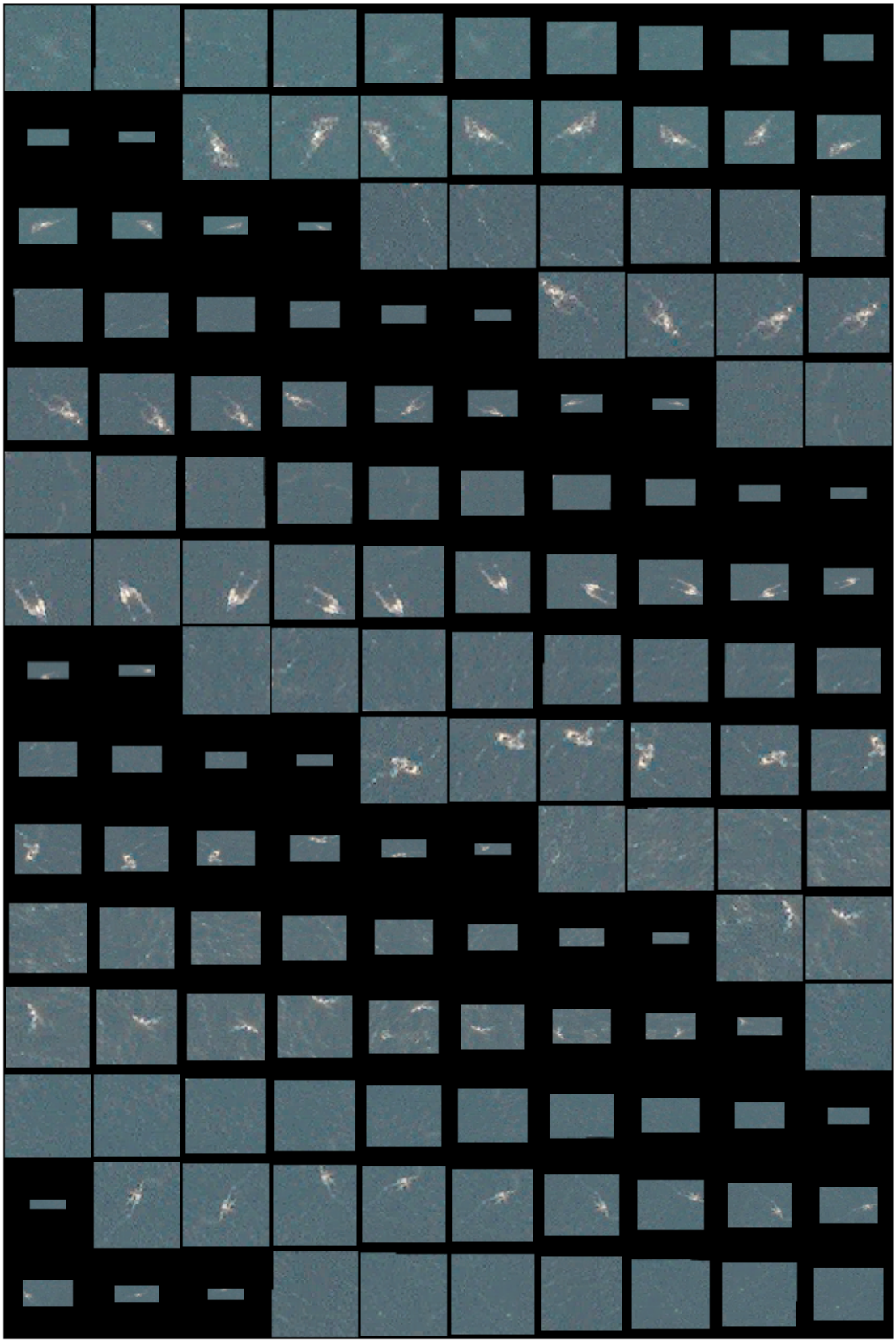
Auckland2011. *Satellite image © 2022 Maxar Technologies.*



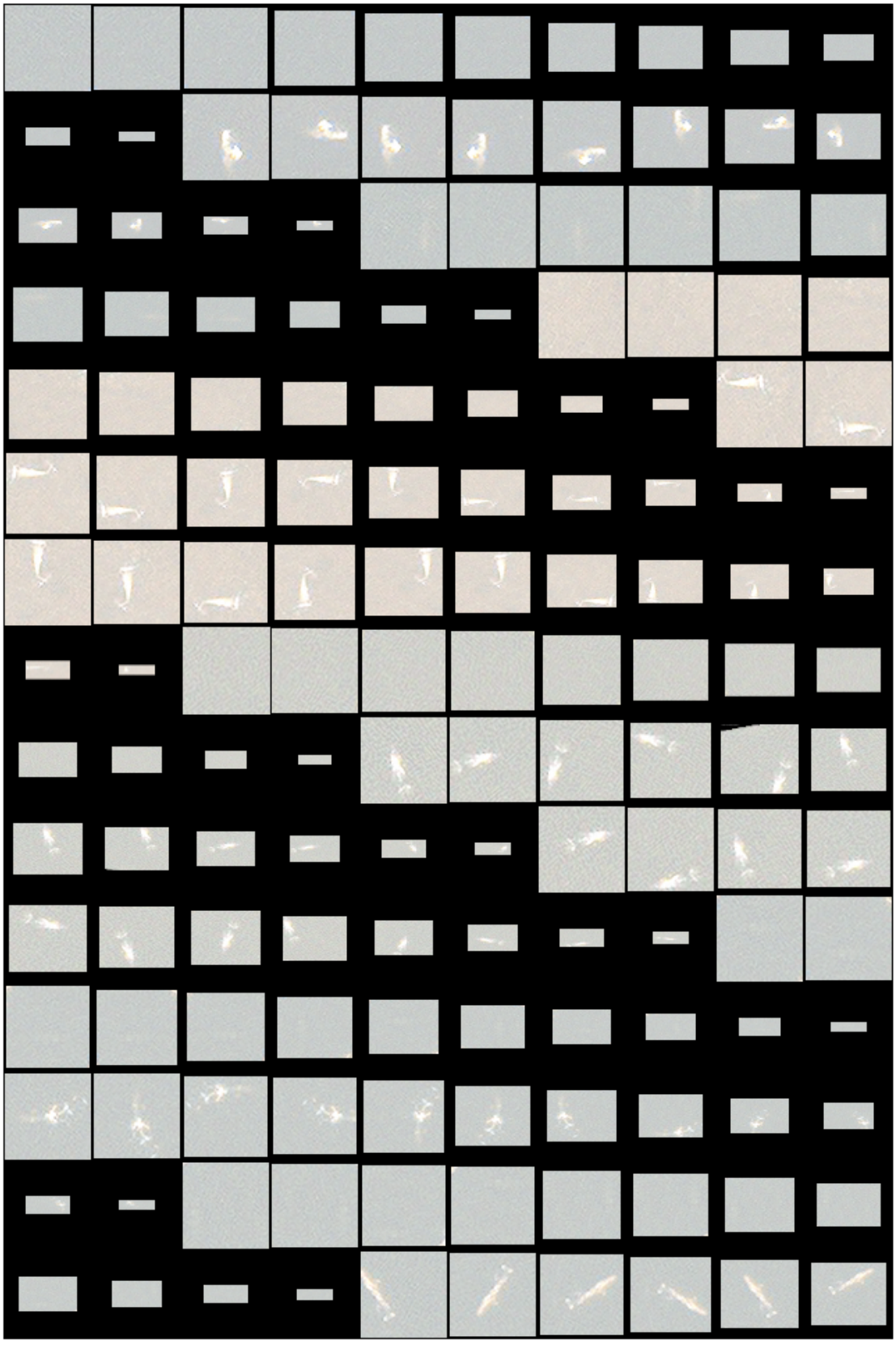
Valdes2016. *Satellite image © 2022 Maxar Technologies.*



Maui2015. *Satellite image © 2022 Maxar Technologies.*



Auckland2006. *Satellite image © 2022 Maxar Technologies.*



Ignacio2017. *Satellite image © 2022 Maxar Technologies.*

E

Appendix E - Hyperparameter Optimization

This appendix documents the hyperparameter optimization described in subsection 5.4.3. It lists the generated training configurations for each tuning stage and reports the corresponding selection results. The appendix is included to make the detector-development process transparent and reproducible beyond the summarized discussion in the main text.

E.1. Generated Hyperparameter Configurations

E.1.1. Stage 1: Exploration

Table E.1: Stage 1 hyperparameter configurations for exploration of the hyperparameter space.

ID	Learning rate	Batch size	IoU crop p	Mosaic p	Resolution
01	10^{-3}	16	0.4	0.4	320
02	10^{-4}	32	0.8	0.0	320
03	10^{-3}	32	0.0	0.0	256
04	10^{-5}	16	0.4	0.8	256
05	10^{-4}	8	0.4	0.0	256
06	10^{-4}	8	0.8	0.8	320
07	10^{-4}	32	0.8	0.4	320
08	10^{-3}	32	0.8	0.8	256
09	10^{-3}	32	0.0	0.0	320
10	10^{-5}	32	0.0	0.8	256

E.1.2. Stage 2: Learning Rate Selection

Table E.2: Stage 2 hyperparameter settings for learning rate optimization.

ID	LR tag	Learning rate	Backbone LR	Batch size	IoU crop p	Mosaic p	Resolution
01	e-40	`1e-4`	`1e-5`	32	0.2	0.2	256
02	e-60	`1e-6`	`1e-7`	32	0.2	0.2	256
03	e-50	`1e-5`	`1e-6`	32	0.2	0.2	256
04	e-35	`1e-3.5`	`1e-4.5`	32	0.2	0.2	256
05	e-55	`1e-5.5`	`1e-6.5`	32	0.2	0.2	256
06	e-45	`1e-4.5`	`1e-5.5`	32	0.2	0.2	256
07	e-30	`1e-3`	`1e-4`	32	0.2	0.2	256

E.1.3. Stage 3: Schedule Refinement

Table E.3: Stage 3 hyperparameter settings for schedule refinement.

ID	Total epochs	Flat epoch	No-aug epoch	Early-stop epoch	Matcher change epoch
01	16	10	4	12	11
02	16	9	6	10	9
03	32	14	12	20	18
04	32	16	8	24	22
05	48	22	12	36	32
06	48	24	8	40	36
07	64	30	12	52	47
08	64	32	8	56	50
09	80	40	12	72	65
10	80	40	8	72	65

E.1.4. Stage 4: Augmentation Global Search

Table E.4: Stage 4 hyperparameter settings for augmentation optimization.

ID	Mosaic p	IoU crop p	Mixup p	CopyBlend p
01	0.0	0.0	0.0	0.0
02	0.0	0.8	0.5	0.5
03	0.8	0.4	0.25	0.25
04	0.4	0.2	0.0	0.0
05	0.2	0.0	0.0	0.5
06	0.0	0.8	0.5	0.25
07	0.4	0.0	0.25	0.0
08	0.2	0.4	0.0	0.25
09	0.8	0.2	0.5	0.5
10	0.8	0.2	0.25	0.0
11	0.0	0.4	0.5	0.5
12	0.4	0.8	0.0	0.25
13	0.2	0.0	0.25	0.0
14	0.4	0.2	0.25	0.0
15	0.8	0.0	0.0	0.25
16	0.2	0.8	0.5	0.5
17	0.0	0.4	0.25	0.5
18	0.0	0.8	0.0	0.25
19	0.4	0.4	0.5	0.0
20	0.2	0.2	0.5	0.25
21	0.8	0.0	0.25	0.5
22	0.2	0.4	0.0	0.0
23	0.4	0.8	0.5	0.0
24	0.0	0.2	0.25	0.5
25	0.8	0.4	0.0	0.25
26	0.8	0.0	0.5	0.25
Counts	0.0: 7 0.2: 6 0.4: 6 0.8: 7	0.0: 7 0.2: 6 0.4: 7 0.8: 6	0.0: 9 0.25: 8 0.5: 9	0.0: 9 0.25: 9 0.5: 8

E.1.5. Stage 5: Augmentation Local Search

Table E.5: Stage 5 hyperparameter settings for augmentation refinement.

ID	Mosaic p	IoU crop p	Mixup p	CopyBlend p
02	0.0	0.8	0.5	0.5
03	0.0	0.4	0.25	0.25
04	0.0	0.8	0.25	0.25
05	0.0	0.4	0.5	0.5
06	0.0	0.8	0.5	0.25
07	0.0	0.4	0.25	0.5
08	0.0	0.4	0.5	0.25
09	0.0	0.8	0.25	0.5

E.2. Results Per Run

E.2.1. Stage 1: Exploration

Table E.6: Stage 1 validation results for exploration of the hyperparameter space.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.798	0.333	0.189	0.960	0.257	23.402
02	0.812	0.336	0.183	0.965	0.238	24.003
03	0.761	0.310	0.170	0.958	0.288	20.999
04	0.793	0.340	0.211	0.964	0.182	21.458
05	0.786	0.336	0.202	0.965	0.193	20.837
06	0.804	0.342	0.211	0.966	0.214	26.341
07	0.807	0.341	0.206	0.967	0.244	25.316
08	0.768	0.315	0.163	0.958	0.211	21.985
09	0.762	0.308	0.168	0.955	0.276	23.963
10	0.792	0.337	0.210	0.965	0.195	21.606

E.2.2. Stage 2: Learning Rate Selection

Table E.7: Stage 2 validation results for fold 1.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.787	0.334	0.212	0.965	0.256	21.495
02	0.658	0.251	0.123	0.953	0.224	22.590
03	0.806	0.340	0.212	0.966	0.224	17.827
04	0.811	0.333	0.191	0.963	0.279	24.939
05	0.761	0.310	0.171	0.964	0.186	29.142
06	0.804	0.333	0.190	0.963	0.213	23.784
07	0.781	0.316	0.167	0.959	0.261	20.966

Table E.8: Stage 2 validation results for fold 2.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.726	0.361	0.324	0.923	0.403	19.368
02	0.590	0.269	0.213	0.886	0.252	26.522
03	0.714	0.338	0.296	0.930	0.297	19.206
04	0.730	0.361	0.327	0.916	0.395	23.348
05	0.696	0.334	0.298	0.916	0.319	19.537
06	0.728	0.355	0.312	0.928	0.345	21.542
07	0.728	0.362	0.323	0.911	0.363	20.183

Table E.9: Stage 2 validation results for fold 3.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.706	0.317	0.244	0.941	0.439	22.728
02	0.587	0.243	0.147	0.894	0.230	18.362
03	0.719	0.324	0.247	0.943	0.406	16.333
04	0.716	0.320	0.244	0.931	0.446	32.316
05	0.698	0.300	0.204	0.925	0.418	17.261
06	0.725	0.323	0.235	0.939	0.389	31.180
07	0.716	0.317	0.218	0.916	0.408	19.826

Table E.10: Stage 2 validation results for fold 4.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.686	0.282	0.172	0.932	0.378	21.529
02	0.558	0.225	0.139	0.909	0.211	17.050
03	0.667	0.282	0.187	0.934	0.320	14.640
04	0.705	0.307	0.210	0.938	0.287	37.382
05	0.638	0.262	0.172	0.928	0.346	17.934
06	0.670	0.279	0.182	0.938	0.300	33.931
07	0.705	0.311	0.227	0.922	0.344	19.709

Table E.11: Mean Stage 2 validation results across folds.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.726	0.323	0.238	0.940	0.369	21.280
02	0.598	0.247	0.156	0.910	0.229	21.131
03	0.727	0.321	0.235	0.943	0.312	17.001
04	0.740	0.330	0.243	0.937	0.352	29.497
05	0.698	0.301	0.211	0.933	0.317	20.968
06	0.732	0.322	0.230	0.942	0.312	27.610
07	0.733	0.327	0.234	0.927	0.344	20.171

E.2.3. Stage 3: Schedule Refinement

Table E.12: Stage 3 validation results for fold 1.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.787	0.337	0.193	0.959	0.256	22.629
02	0.790	0.341	0.223	0.961	0.165	26.807
03	0.798	0.336	0.200	0.963	0.223	18.419
04	0.789	0.340	0.214	0.962	0.177	25.107
05	0.797	0.340	0.207	0.962	0.301	21.460
06	0.802	0.339	0.210	0.965	0.218	16.970
07	0.787	0.336	0.213	0.961	0.304	24.587
08	0.797	0.335	0.201	0.966	0.248	19.955
09	0.787	0.334	0.212	0.965	0.256	21.495
10	0.792	0.336	0.205	0.962	0.227	29.649

Table E.13: Stage 3 validation results for fold 2.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.712	0.348	0.304	0.922	0.357	27.713
02	0.704	0.352	0.321	0.917	0.355	24.508
03	0.720	0.358	0.319	0.921	0.361	18.941
04	0.704	0.348	0.300	0.922	0.343	24.207
05	0.727	0.360	0.315	0.923	0.323	19.963
06	0.700	0.350	0.319	0.918	0.368	16.539
07	0.721	0.358	0.315	0.924	0.369	23.835
08	0.732	0.369	0.342	0.924	0.334	19.608
09	0.726	0.361	0.324	0.923	0.403	19.368
10	0.724	0.367	0.339	0.921	0.351	24.955

Table E.14: Stage 3 validation results for fold 3.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.751	0.334	0.248	0.938	0.411	27.121
02	0.744	0.333	0.252	0.934	0.392	25.008
03	0.731	0.324	0.239	0.935	0.447	16.667
04	0.753	0.336	0.249	0.942	0.419	24.224
05	0.731	0.324	0.242	0.937	0.429	19.465
06	0.719	0.323	0.232	0.944	0.432	15.640
07	0.732	0.324	0.239	0.941	0.416	22.814
08	0.738	0.329	0.244	0.937	0.434	18.193
09	0.706	0.317	0.244	0.941	0.439	22.728
10	0.734	0.329	0.250	0.938	0.431	21.429

Table E.15: Stage 3 validation results for fold 4.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.694	0.294	0.201	0.937	0.312	23.421
02	0.690	0.289	0.203	0.934	0.298	26.686
03	0.672	0.281	0.180	0.937	0.397	17.509
04	0.703	0.290	0.198	0.937	0.377	23.769
05	0.686	0.286	0.186	0.933	0.323	17.706
06	0.686	0.285	0.185	0.933	0.368	15.797
07	0.701	0.290	0.190	0.938	0.351	22.495
08	0.687	0.283	0.186	0.934	0.350	18.295
09	0.686	0.282	0.172	0.932	0.378	21.529
10	0.691	0.287	0.185	0.932	0.380	22.297

Table E.16: Mean Stage 3 validation results across folds.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.736	0.328	0.236	0.939	0.334	25.221
02	0.732	0.329	0.250	0.937	0.303	25.752
03	0.730	0.325	0.235	0.939	0.357	17.884
04	0.737	0.328	0.240	0.941	0.329	24.327
05	0.735	0.328	0.237	0.939	0.344	19.648
06	0.727	0.324	0.237	0.940	0.346	16.237
07	0.735	0.327	0.239	0.941	0.360	23.433
08	0.739	0.329	0.243	0.940	0.342	19.013
09	0.726	0.323	0.238	0.940	0.369	21.280
10	0.735	0.330	0.245	0.938	0.347	24.582

E.2.4. Stage 4: Augmentation Global Search

Table E.17: Stage 4 validation results for fold 1.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.794	0.330	0.183	0.965	0.260	32.832
02	0.808	0.339	0.188	0.960	0.247	61.279
03	0.791	0.333	0.197	0.960	0.238	76.992
04	0.780	0.327	0.188	0.963	0.223	48.011
05	0.798	0.340	0.214	0.964	0.223	54.281
06	0.793	0.327	0.198	0.963	0.286	76.913
07	0.781	0.332	0.199	0.961	0.219	82.431
08	0.779	0.332	0.208	0.963	0.178	64.099
09	0.800	0.333	0.203	0.964	0.237	82.535
10	0.789	0.328	0.182	0.962	0.252	28.857
11	0.795	0.336	0.198	0.964	0.217	34.397
12	0.787	0.334	0.205	0.959	0.227	70.095
13	0.784	0.321	0.189	0.960	0.240	70.064
14	0.781	0.331	0.203	0.965	0.262	83.625
15	0.810	0.338	0.202	0.962	0.205	64.366
16	0.788	0.336	0.199	0.965	0.225	56.118
17	0.807	0.342	0.206	0.964	0.204	60.066
18	0.797	0.334	0.202	0.963	0.233	48.433
19	0.801	0.340	0.211	0.967	0.258	27.070
20	0.792	0.333	0.205	0.963	0.180	28.534
21	0.803	0.340	0.216	0.963	0.219	20.817
22	0.780	0.331	0.195	0.962	0.186	29.482
23	0.794	0.336	0.210	0.966	0.241	21.910
24	0.797	0.341	0.199	0.965	0.213	34.776
25	0.778	0.330	0.205	0.964	0.221	21.811
26	0.785	0.338	0.209	0.961	0.238	40.069

Table E.18: Stage 4 validation results for fold 2.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.682	0.335	0.303	0.915	0.357	31.566
02	0.695	0.332	0.291	0.913	0.346	31.857
03	0.726	0.354	0.313	0.919	0.352	11.645
04	0.692	0.327	0.285	0.912	0.344	25.391
05	0.735	0.363	0.324	0.920	0.354	21.443
06	0.730	0.347	0.305	0.917	0.341	31.592
07	0.695	0.343	0.311	0.908	0.345	20.598
08	0.737	0.362	0.326	0.921	0.349	31.665
09	0.721	0.344	0.296	0.921	0.387	31.621
10	0.706	0.340	0.306	0.910	0.322	17.998
11	0.731	0.355	0.320	0.920	0.385	31.698
12	0.713	0.353	0.324	0.912	0.368	31.552
13	0.684	0.325	0.273	0.911	0.369	32.219
14	0.717	0.344	0.294	0.918	0.341	17.922
15	0.733	0.354	0.311	0.916	0.381	13.102
16	0.715	0.345	0.304	0.913	0.335	24.706
17	0.724	0.354	0.303	0.918	0.351	31.470
18	0.698	0.338	0.300	0.911	0.342	14.041
19	0.719	0.353	0.320	0.920	0.381	11.489
20	0.724	0.350	0.310	0.915	0.337	18.693
21	0.716	0.352	0.320	0.920	0.353	13.967
22	0.696	0.340	0.303	0.911	0.368	31.412
23	0.702	0.349	0.304	0.914	0.347	10.831
24	0.724	0.344	0.293	0.916	0.370	28.605
25	0.717	0.347	0.308	0.917	0.354	13.658
26	0.716	0.352	0.311	0.921	0.351	37.357

Table E.19: Stage 4 validation results for fold 3.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.753	0.337	0.253	0.932	0.439	37.719
02	0.728	0.325	0.242	0.934	0.412	32.104
03	0.716	0.324	0.241	0.932	0.432	19.842
04	0.739	0.323	0.236	0.934	0.432	18.940
05	0.746	0.338	0.252	0.936	0.419	18.899
06	0.753	0.334	0.240	0.934	0.451	37.319
07	0.721	0.323	0.240	0.934	0.426	18.619
08	0.744	0.331	0.248	0.936	0.458	37.309
09	0.742	0.330	0.240	0.937	0.440	37.668
10	0.741	0.332	0.247	0.934	0.413	21.976
11	0.755	0.342	0.250	0.936	0.390	32.034
12	0.730	0.326	0.241	0.932	0.430	32.098
13	0.747	0.332	0.243	0.937	0.417	37.650
14	0.735	0.327	0.244	0.932	0.429	26.808
15	0.747	0.335	0.250	0.937	0.403	37.237
16	0.758	0.337	0.244	0.939	0.399	29.866
17	0.739	0.325	0.232	0.937	0.421	32.116
18	0.731	0.328	0.249	0.932	0.443	32.038
19	0.744	0.336	0.241	0.938	0.437	19.824
20	0.720	0.324	0.229	0.930	0.439	24.621
21	0.735	0.324	0.231	0.933	0.419	19.568
22	0.715	0.324	0.235	0.933	0.422	32.018
23	0.734	0.327	0.239	0.936	0.451	27.811
24	0.725	0.325	0.233	0.938	0.372	32.709
25	0.736	0.326	0.232	0.933	0.412	19.691
26	0.732	0.329	0.238	0.941	0.398	29.029

Table E.20: Stage 4 validation results for fold 4.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.705	0.294	0.189	0.931	0.328	56.315
02	0.714	0.304	0.210	0.934	0.310	40.875
03	0.682	0.283	0.176	0.934	0.318	22.285
04	0.707	0.290	0.189	0.933	0.348	22.370
05	0.715	0.303	0.193	0.934	0.324	22.990
06	0.707	0.288	0.177	0.931	0.358	56.306
07	0.684	0.286	0.179	0.930	0.317	24.574
08	0.713	0.295	0.186	0.932	0.366	56.360
09	0.695	0.286	0.178	0.935	0.348	56.398
10	0.688	0.288	0.186	0.930	0.316	26.122
11	0.708	0.302	0.195	0.936	0.331	40.804
12	0.687	0.287	0.177	0.930	0.335	40.897
13	0.704	0.291	0.191	0.932	0.308	56.372
14	0.698	0.293	0.188	0.936	0.349	26.237
15	0.697	0.291	0.185	0.933	0.359	56.335
16	0.710	0.295	0.185	0.935	0.328	48.133
17	0.700	0.286	0.179	0.937	0.344	40.962
18	0.692	0.288	0.188	0.934	0.339	40.993
19	0.707	0.300	0.190	0.935	0.342	22.178
20	0.697	0.287	0.179	0.930	0.344	28.252
21	0.703	0.289	0.192	0.933	0.341	22.076
22	0.689	0.288	0.181	0.932	0.342	40.999
23	0.688	0.288	0.190	0.931	0.318	26.607
24	0.685	0.287	0.191	0.932	0.418	33.857
25	0.694	0.298	0.186	0.932	0.355	11.757
26	0.688	0.281	0.179	0.932	0.326	22.730

Table E.21: Mean Stage 4 validation results across folds.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
01	0.734	0.324	0.232	0.936	0.346	39.608
02	0.736	0.325	0.233	0.935	0.329	41.529
03	0.729	0.324	0.232	0.936	0.335	32.691
04	0.730	0.317	0.225	0.936	0.337	28.678
05	0.749	0.336	0.246	0.939	0.330	29.403
06	0.746	0.324	0.230	0.936	0.359	50.533
07	0.720	0.321	0.232	0.933	0.327	36.555
08	0.743	0.330	0.242	0.938	0.338	47.358
09	0.739	0.323	0.229	0.939	0.353	52.056
10	0.731	0.322	0.230	0.934	0.326	23.738
11	0.747	0.334	0.241	0.939	0.331	34.733
12	0.729	0.325	0.237	0.933	0.340	43.660
13	0.730	0.317	0.224	0.935	0.334	49.076
14	0.733	0.324	0.232	0.938	0.345	38.648
15	0.747	0.330	0.237	0.937	0.337	42.760
16	0.743	0.328	0.233	0.938	0.322	39.706
17	0.742	0.327	0.230	0.939	0.330	41.154
18	0.730	0.322	0.235	0.935	0.339	33.876
19	0.743	0.332	0.240	0.940	0.354	20.140
20	0.733	0.323	0.231	0.935	0.325	25.025
21	0.739	0.326	0.240	0.937	0.333	19.107
22	0.720	0.321	0.229	0.935	0.329	33.478
23	0.730	0.325	0.236	0.937	0.339	21.790
24	0.732	0.324	0.231	0.938	0.343	32.487
25	0.732	0.325	0.233	0.934	0.335	16.729
26	0.730	0.325	0.234	0.939	0.328	32.296

E.2.5. Stage 5: Augmentation Local Search

Table E.22: Stage 5 validation results for fold 1.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
02	0.804	0.339	0.208	0.963	0.257	16.139
03	0.789	0.332	0.200	0.962	0.294	39.382
04	0.784	0.333	0.201	0.959	0.237	34.007
05	0.793	0.334	0.196	0.964	0.194	16.524
06	0.805	0.333	0.193	0.966	0.197	31.511
07	0.808	0.348	0.215	0.967	0.244	21.661
08	0.790	0.330	0.186	0.962	0.274	34.512
09	0.796	0.333	0.198	0.965	0.196	55.620

Table E.23: Stage 5 validation results for fold 2.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
02	0.714	0.355	0.320	0.917	0.312	18.915
03	0.723	0.356	0.304	0.920	0.389	63.676
04	0.711	0.348	0.306	0.924	0.349	42.504
05	0.730	0.364	0.311	0.923	0.315	21.599
06	0.728	0.360	0.315	0.918	0.357	19.618
07	0.711	0.347	0.308	0.915	0.315	14.913
08	0.727	0.360	0.308	0.916	0.337	35.231
09	0.724	0.361	0.318	0.911	0.399	23.179

Table E.24: Stage 5 validation results for fold 3.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
02	0.721	0.320	0.233	0.935	0.433	15.837
03	0.718	0.317	0.223	0.933	0.443	32.948
04	0.723	0.320	0.242	0.938	0.454	48.600
05	0.722	0.320	0.237	0.935	0.458	14.932
06	0.716	0.316	0.228	0.938	0.444	17.616
07	0.724	0.317	0.234	0.936	0.423	13.484
08	0.722	0.317	0.226	0.932	0.453	39.317
09	0.737	0.323	0.238	0.935	0.394	15.992

Table E.25: Stage 5 validation results for fold 4.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
02	0.687	0.281	0.179	0.929	0.362	13.907
03	0.670	0.276	0.177	0.915	0.380	41.542
04	0.680	0.280	0.174	0.928	0.348	49.173
05	0.698	0.293	0.192	0.933	0.348	13.331
06	0.680	0.279	0.173	0.924	0.361	17.968
07	0.690	0.284	0.182	0.933	0.373	13.096
08	0.685	0.285	0.184	0.931	0.362	55.187
09	0.692	0.297	0.213	0.932	0.310	16.625

Table E.26: Mean Stage 5 validation results across folds.

ID	AP50	AP50:95	AP75	Best F1	F1 thr	Latency [ms/img]
02	0.732	0.324	0.235	0.936	0.341	16.199
03	0.725	0.320	0.226	0.933	0.377	44.387
04	0.724	0.320	0.231	0.937	0.347	43.571
05	0.736	0.328	0.234	0.939	0.329	16.597
06	0.732	0.322	0.227	0.936	0.340	21.678
07	0.733	0.324	0.235	0.938	0.339	15.788
08	0.731	0.323	0.226	0.935	0.356	41.062
09	0.737	0.328	0.242	0.936	0.324	27.854

E.3. Model Size Comparison

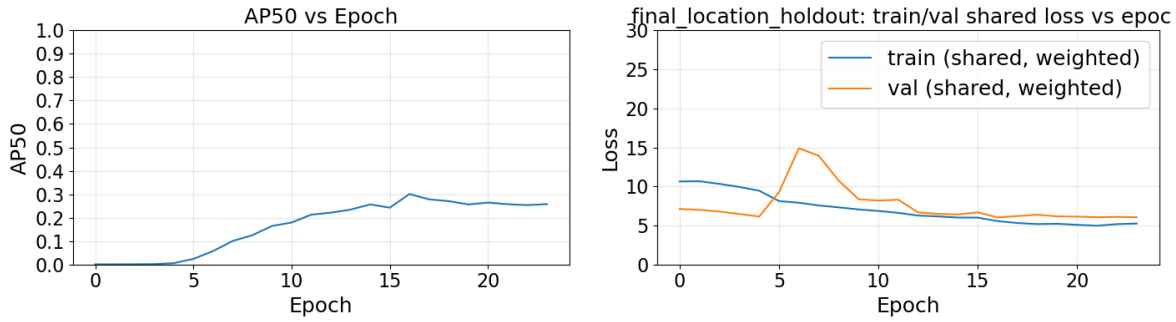


Figure E.1: Validation AP_{50} with the training and validation loss for the 24-epoch Atto model. Note that the schedule and augmentation parameters were optimized for model size S only.

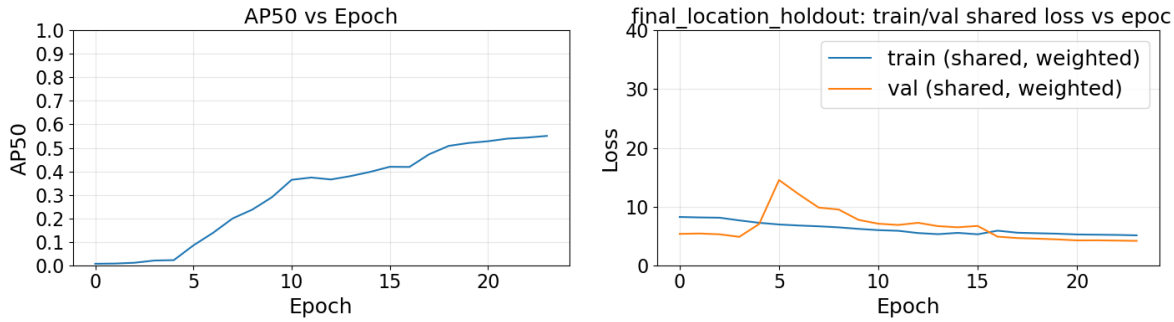


Figure E.2: Validation AP_{50} with the training and validation loss for the 24-epoch Femto model. Note that the schedule and augmentation parameters were optimized for model size S only.

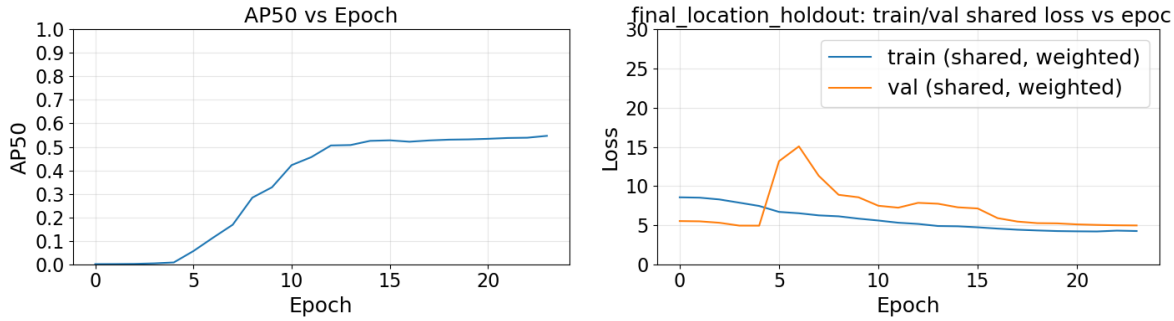


Figure E.3: Validation AP_{50} with the training and validation loss for the 24-epoch Pico model. Note that the schedule and augmentation parameters were optimized for model size S only.

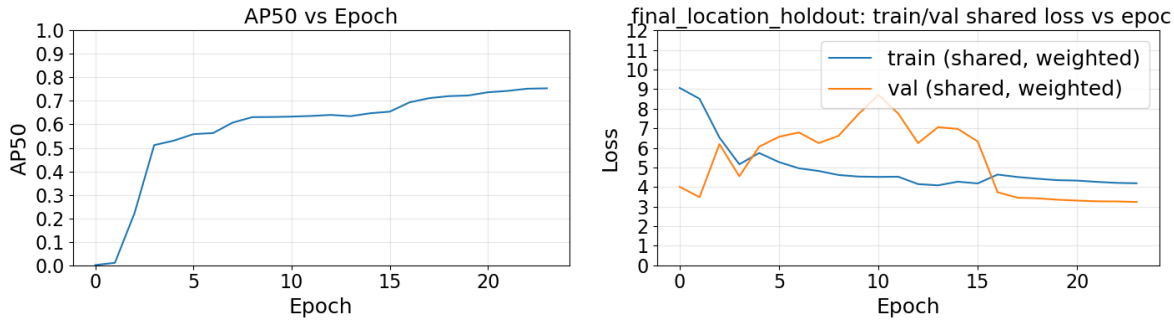


Figure E.4: Validation AP_{50} with the training and validation loss for the 24-epoch N model. Note that the schedule and augmentation parameters were optimized for model size S only.

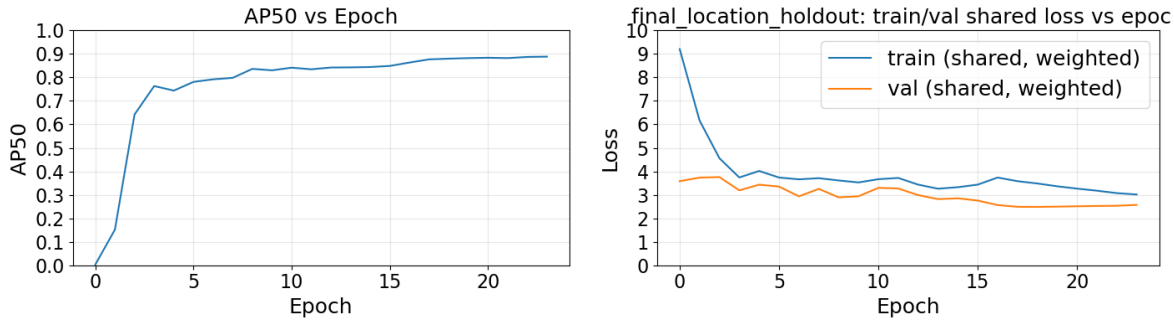


Figure E.5: Validation AP_{50} with the training and validation loss for the 24-epoch S model. Note that the schedule and augmentation parameters were optimized for this model only.

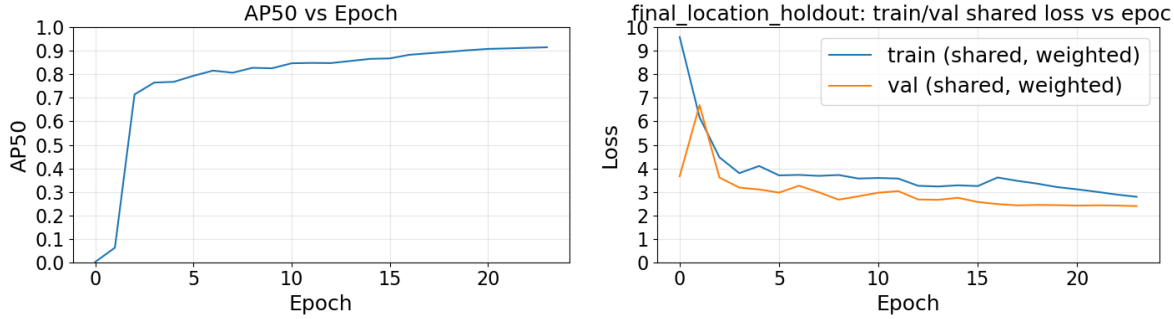


Figure E.6: Validation AP_{50} with the training and validation loss for the 24-epoch M model. Note that the schedule and augmentation parameters were optimized for model size S only.

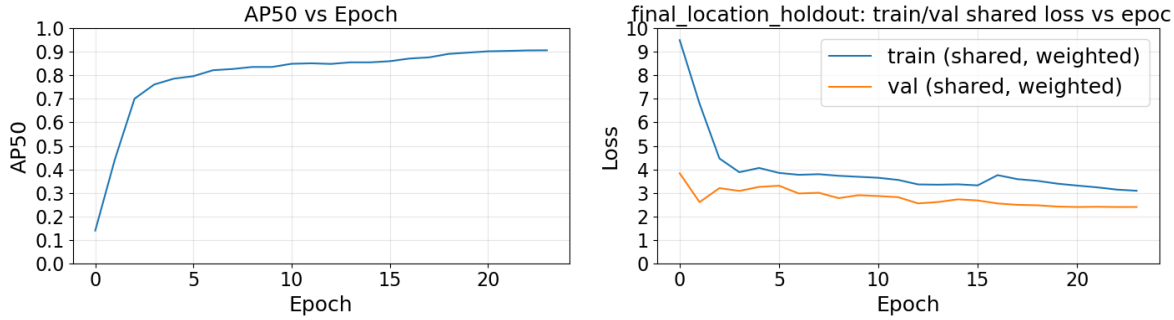


Figure E.7: Validation AP_{50} with the training and validation loss for the 24-epoch L model. Note that the schedule and augmentation parameters were optimized for model size S only.

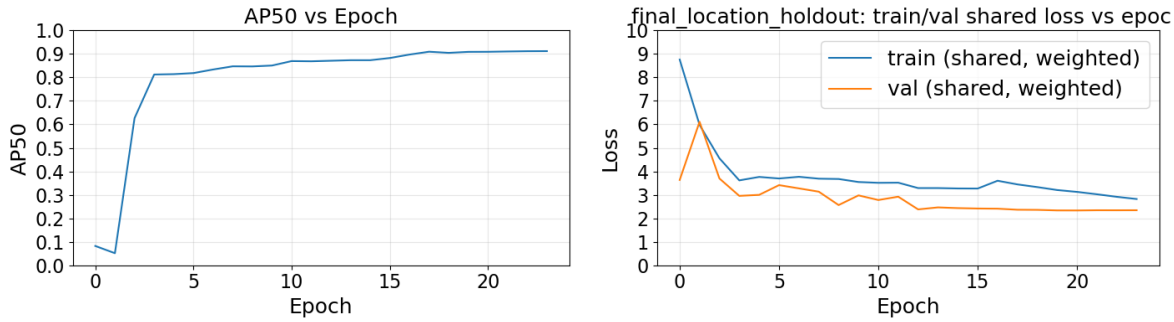


Figure E.8: Validation AP_{50} with the training and validation loss for the 24-epoch X model. Note that the schedule and augmentation parameters were optimized for model size S only.

E.4. Final Model Training

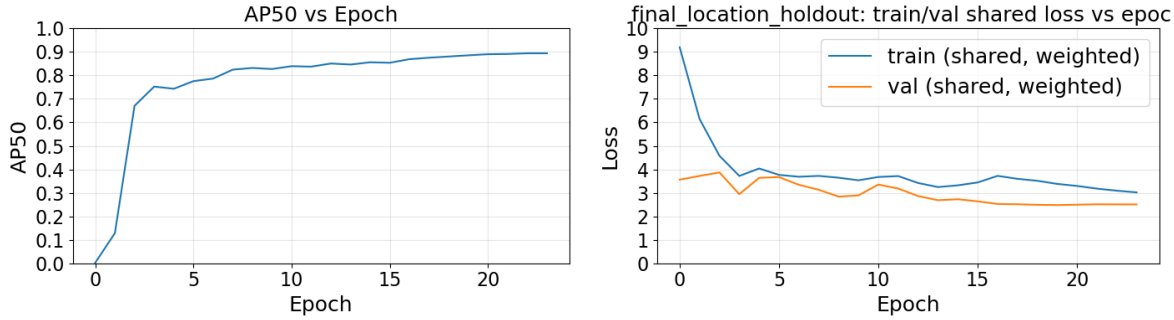


Figure E.9: Validation AP_{50} with the training and validation loss for model including geometric and radiometric effects. Note that the schedule and augmentation parameters were optimized for this model only.

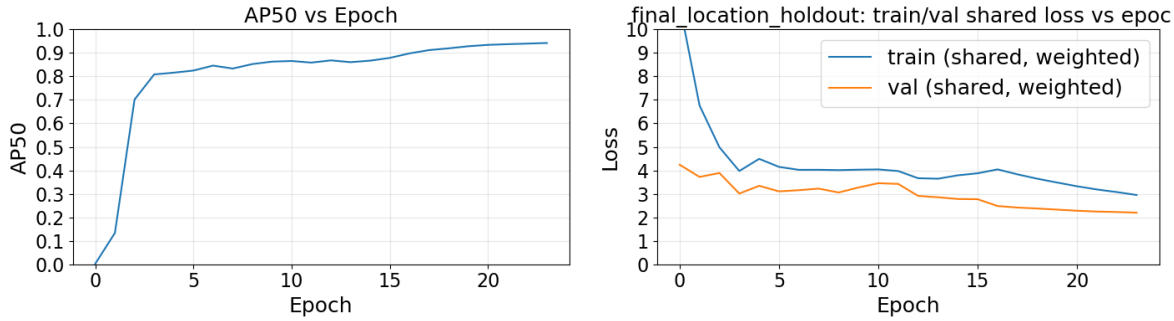


Figure E.10: Validation AP_{50} with the training and validation loss for model including radiometric effects only, trained on the radiometric nadir images. Note that the schedule and augmentation parameters were optimized for the model including both radiometric and geometric effects.

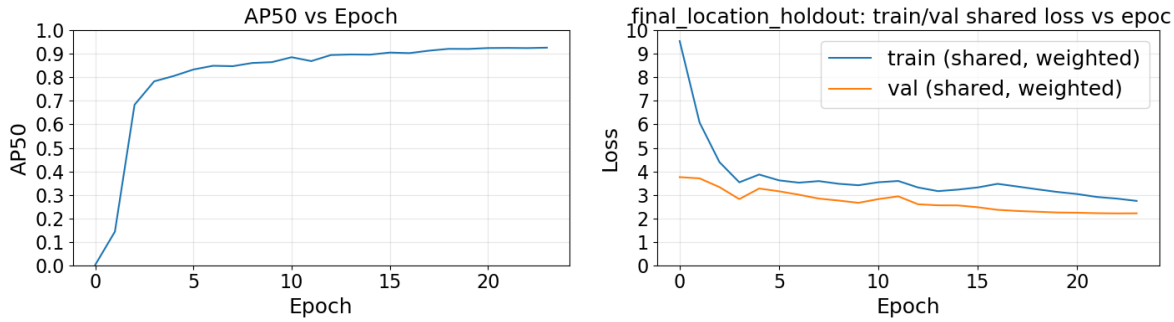


Figure E.11: Validation AP_{50} with the training and validation loss for model including geometric effects only, trained on the input patch images translated off-nadir. Note that the schedule and augmentation parameters were optimized for the model including both radiometric and geometric effects.

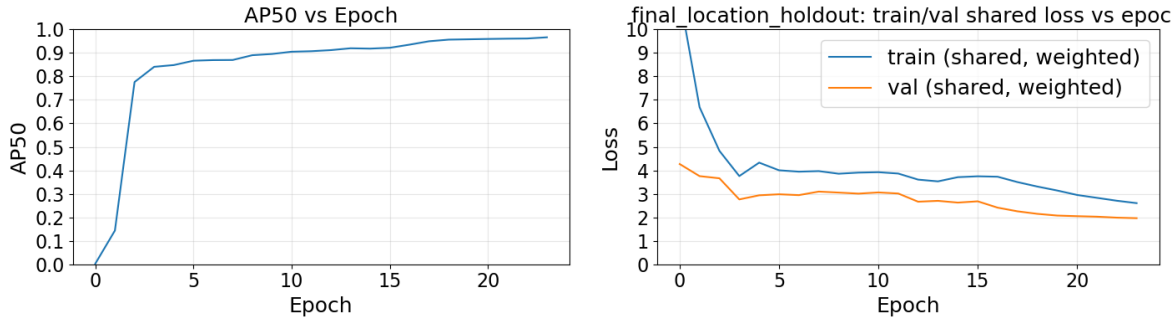


Figure E.12: Validation AP_{50} with the training and validation loss for model without radiometric and geometric effects, trained on the original cropped input patches. Note that the schedule and augmentation parameters were optimized for the model including both radiometric and geometric effects.

E.5. Validation Crossfold Split

Table E.27: Cross-validation split definition showing training, validation, and test locations for fold 1-12.

Fold	Train locations	Validation locations	Test locations
Fold 1	Ignacio2017, Maui2015 Valdes2012, Valdes2014 Valdes2016	Auckland2011 Witsand2009	Pelagos2016 Auckland2006
Fold 2	Auckland2011, Valdes2012 Valdes2014, Valdes2016 Witsand2009	Ignacio2017 Maui2015	Pelagos2016 Auckland2006
Fold 3	Auckland2011, Ignacio2017 Maui2015, Valdes2012 Witsand2009	Valdes2014 Valdes2016	Pelagos2016 Auckland2006
Fold 4	Auckland2011, Ignacio2017 Maui2015, Valdes2014 Witsand2009	Valdes2012 Valdes2016	Pelagos2016 Auckland2006
Fold 5	Ignacio2017, Maui2015 Valdes2012, Valdes2016 Witsand2009	Auckland2011 Valdes2014	Pelagos2016 Auckland2006
Fold 6	Auckland2011, Maui2015 Valdes2012, Valdes2014 Valdes2016	Ignacio2017 Witsand2009	Pelagos2016 Auckland2006
Fold 7	Auckland2011, Ignacio2017 Valdes2014, Valdes2016 Witsand2009	Maui2015 Valdes2012	Pelagos2016 Auckland2006
Fold 8	Auckland2011, Ignacio2017 Maui2015, Valdes2016 Witsand2009	Valdes2012 Valdes2014	Pelagos2016 Auckland2006
Fold 9	Ignacio2017, Maui2015 Valdes2012, Valdes2014 Witsand2009	Auckland2011 Valdes2016	Pelagos2016 Auckland2006
Fold 10	Auckland2011, Ignacio2017 Valdes2012, Valdes2014 Valdes2016	Maui2015 Witsand2009	Pelagos2016 Auckland2006
Fold 11	Auckland2011, Maui2015 Valdes2012, Valdes2014 Witsand2009	Ignacio2017 Valdes2016	Pelagos2016 Auckland2006
Fold 12	Maui2015, Valdes2012 Valdes2014, Valdes2016 Witsand2009	Auckland2011 Ignacio2017	Pelagos2016 Auckland2006

Table E.28: Cross-validation split definition showing training, validation, and test locations for fold 13-21.

Fold	Train locations	Validation locations	Test locations
Fold 13	Auckland2011, Ignacio2017 Valdes2012, Valdes2016 Witsand2009	Maui2015 Valdes2014	Pelagos2016 Auckland2006
Fold 14	Auckland2011, Ignacio2017 Maui2015, Valdes2014 Valdes2016	Valdes2012 Witsand2009	Pelagos2016 Auckland2006
Fold 15	Ignacio2017, Valdes2012 Valdes2014, Valdes2016 Witsand2009	Auckland2011 Maui2015	Pelagos2016 Auckland2006
Fold 16	Auckland2011, Maui2015 Valdes2012, Valdes2016 Witsand2009	Ignacio2017 Valdes2014	Pelagos2016 Auckland2006
Fold 17	Auckland2011, Ignacio2017 Maui2015, Valdes2012 Valdes2014	Valdes2016 Witsand2009	Pelagos2016 Auckland2006
Fold 18	Ignacio2017, Maui2015 Valdes2014, Valdes2016 Witsand2009	Auckland2011 Valdes2012	Pelagos2016 Auckland2006
Fold 19	Auckland2011, Ignacio2017 Maui2015, Valdes2012 Valdes2016	Valdes2014 Witsand2009	Pelagos2016 Auckland2006
Fold 20	Auckland2011, Ignacio2017 Valdes2012, Valdes2014 Witsand2009	Maui2015 Valdes2016	Pelagos2016 Auckland2006
Fold 21	Auckland2011, Maui2015 Valdes2014, Valdes2016 Witsand2009	Ignacio2017 Valdes2012	Pelagos2016 Auckland2006

Appendix F - Off Nadir Imaging Rendering Pipeline Validation

This appendix provides the detailed numerical results underlying the off-nadir rendering validation presented in section 6.2. The generated images are compared against two Φ-Sat-2 L1 products, and the appendix reports the reflectance statistics for the R, G, and B bands per input texture, their averages across locations, and the absolute and percentage differences with respect to the reference observations. In addition, the reflectance curves as a function of wind speed are included to show how the rendering output varies with the assumed sea-surface roughness.

F.1. A81 Φ-Sat-2 Data

F.1.1. Reflectance Tables

Table F.1: Reflectance statistics for the R spectral band comparing generated images for the five input textures with the Φ-Sat-2 A81 reference observation at an estimated wind speed of 7.8 m/s.

Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	245000	0.071885	0.205913	0.130674	0.093411	0.133724	0.174607
Auckland 2011	Generated	245000	0.093288	0.177086	0.128622	0.109618	0.128160	0.151741
Maui 2015	Generated	245000	0.023449	0.397269	0.135766	0.109516	0.135014	0.187187
Pelagos 2016	Generated	245000	0.068790	0.191325	0.137561	0.121442	0.137570	0.153484
Valdes 2014	Generated	245000	0.022609	0.895852	0.124899	0.043031	0.126148	0.294199
Mean	Generated	245000	0.056004	0.373489	0.131504	0.095404	0.132123	0.192244
Φ-Sat-2	Original	262144	0.0353	0.3603	0.141024	0.0684	0.1385	0.2398
Absolute difference	Generated vs. Φ-Sat-2	17144	0.020704	0.013189	0.009520	0.027004	0.006377	0.047556
Percentage difference (%)	Generated vs. Φ-Sat-2	6.54	58.65	3.66	6.75	39.48	**4.60**	19.83

Table F.2: Reflectance statistics for the G spectral band comparing generated images for the five input textures with the Φ-Sat-2 A81 reference observation at an estimated wind speed of 7.8 m/s.

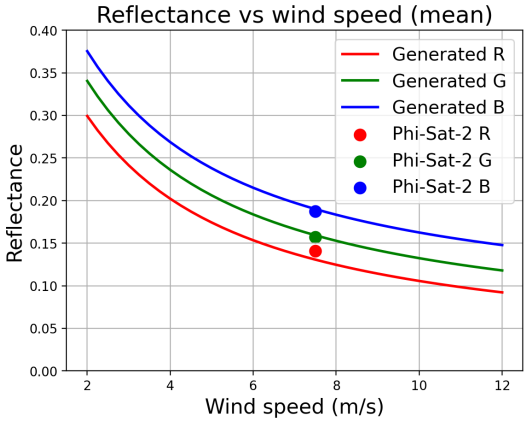
Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	245000	0.091859	0.231442	0.159277	0.116921	0.158419	0.200957
Auckland 2011	Generated	245000	0.114629	0.211184	0.160977	0.136014	0.160628	0.187850
Maui 2015	Generated	245000	0.027336	0.477437	0.157499	0.127809	0.156982	0.212602
Pelagos 2016	Generated	245000	0.081803	0.217378	0.157519	0.139586	0.157425	0.175590
Valdes 2014	Generated	245000	0.026681	0.940441	0.202038	0.051850	0.152200	0.586349
Mean	Generated	245000	0.068462	0.415577	0.167462	0.114436	0.157131	0.272670
Φ-Sat-2	Original	262144	0.0483	0.3250	0.157100	0.0887	0.1552	0.2441
Absolute difference	Generated vs. Φ-Sat-2	17144	0.020162	0.090577	0.010361	0.025736	0.001931	0.028570
Percentage difference (%)	Generated vs. Φ-Sat-2	6.54	41.74	27.87	6.60	29.02	**1.24**	11.70

Table F.3: Reflectance statistics for the B spectral band comparing generated images for the five input textures with the Φ-Sat-2 A81 reference observation at an estimated wind speed of 7.8 m/s.

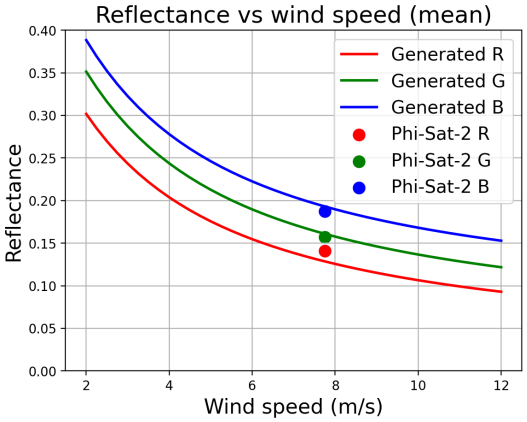
Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	245000	0.110442	0.273988	0.190039	0.140700	0.190050	0.239785
Auckland 2011	Generated	245000	0.138455	0.255212	0.193100	0.163947	0.192452	0.224200
Maui 2015	Generated	245000	0.031848	0.563671	0.187566	0.148914	0.186483	0.249841
Pelagos 2016	Generated	245000	0.092531	0.265764	0.192114	0.171505	0.191860	0.213804
Valdes 2014	Generated	245000	0.082415	0.816400	0.194975	0.125691	0.183695	0.324576
Mean	Generated	245000	0.091138	0.435007	0.191559	0.150151	0.188908	0.250441
Φ-Sat-2	Original	262144	0.0885	0.3427	0.187654	0.1298	0.1865	0.2558
Absolute difference	Generated vs. Φ-Sat-2	17144	0.002638	0.092307	0.003904	0.020351	0.002408	0.005359
Percentage difference (%)	Generated vs. Φ-Sat-2	6.54	2.98	26.93	2.08	15.68	**1.29**	2.09

F.1.2. Reflectance Wind Velocity Match

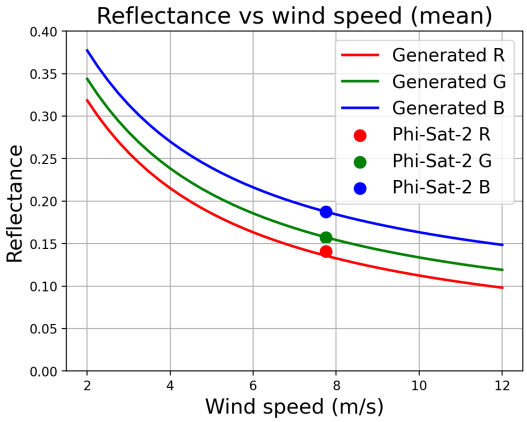
Per Image



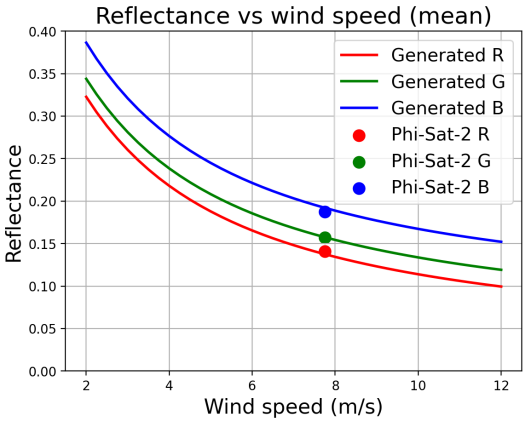
(a) Auckland 2006



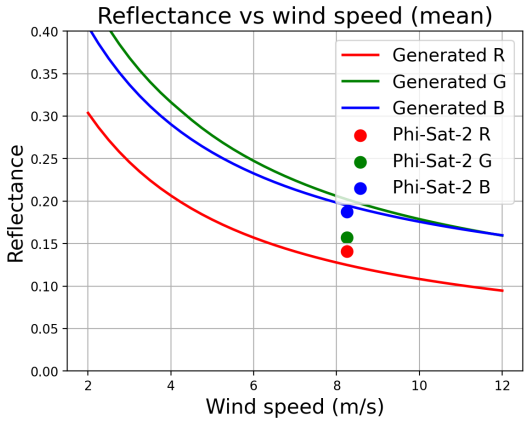
(b) Auckland 2011



(c) Maui 2015



(d) Pelagos 2016



(e) Valdes 2014

Figure F.1: Mean reflectance as a function of wind speed for the A81 observation geometry for five ocean input textures. The curves represent generated reflectance values for the R, G, and B spectral bands, while the markers indicate the true reflectance of the Φ-Sat-2 observation for each band.

Per Band

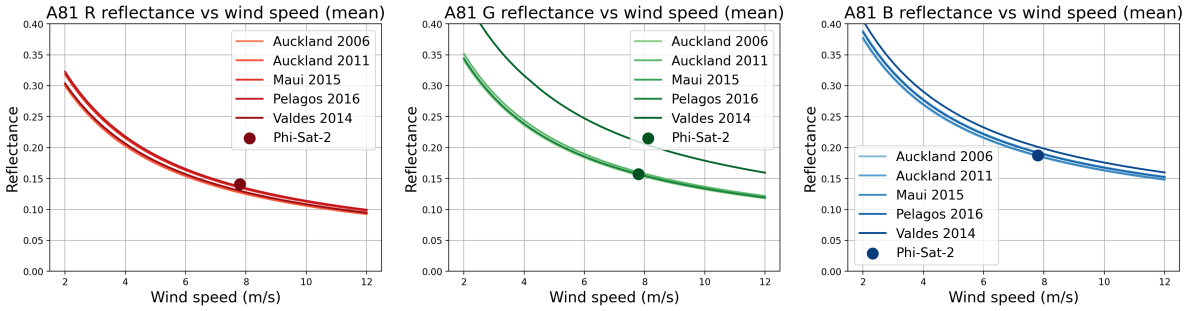


Figure F.2: Mean reflectance as a function of wind speed for the A81 observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

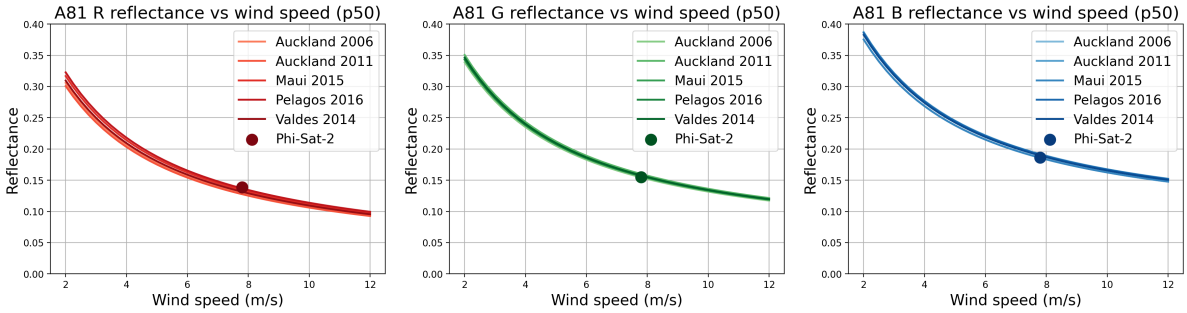


Figure F.3: Median reflectance ($p50$) as a function of wind speed for the A81 observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

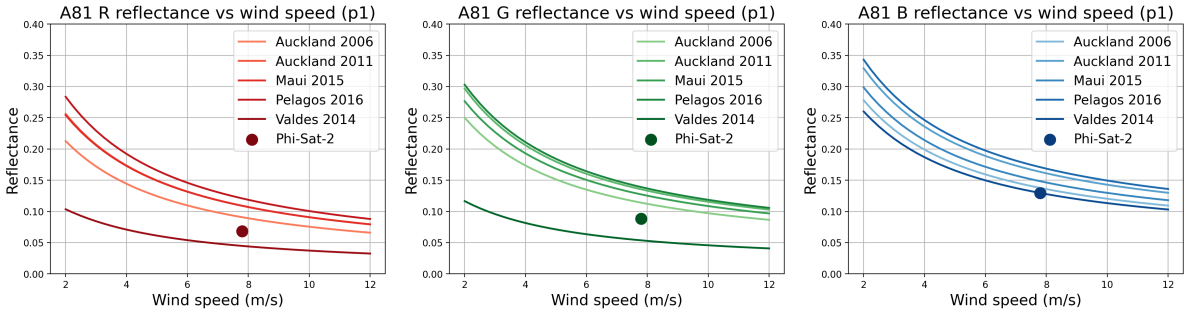


Figure F.4: Lower reflectance percentile ($p1$) as a function of wind speed for the A81 observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

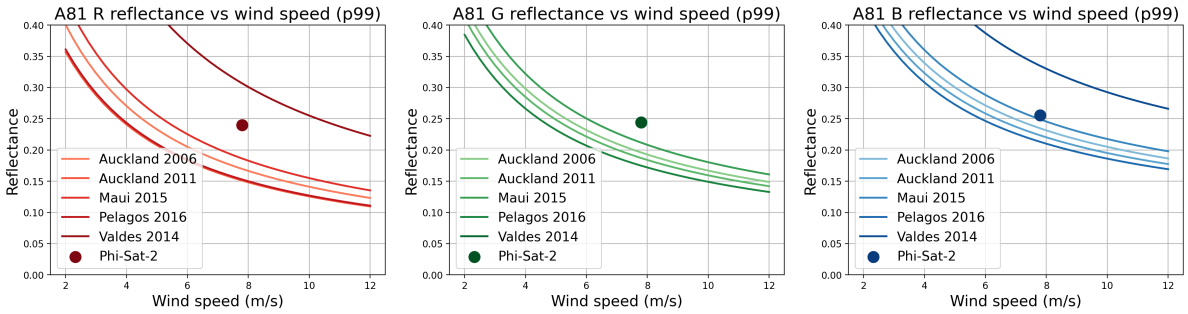


Figure F.5: Upper reflectance percentile ($p99$) as a function of wind speed for the A81 observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

F.2. C3E Φ-Sat-2 Data

F.2.1. Reflectance Tables

Table F.4: Reflectance statistics for the R spectral band comparing generated images for the five input textures with the Φ-Sat-2 C3E reference observation at an estimated wind speed of 2 m/s.

Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	253000	0.008986	0.025400	0.016419	0.011749	0.016809	0.021937
Auckland 2011	Generated	253000	0.012180	0.022796	0.016553	0.014150	0.016507	0.019478
Maui 2015	Generated	253000	0.002818	0.051723	0.017474	0.014331	0.017370	0.024016
Pelagos 2016	Generated	253000	0.009049	0.024622	0.017705	0.015732	0.017699	0.019715
Valdes 2014	Generated	253000	0.002937	0.173437	0.016855	0.005716	0.016996	0.039598
Mean	Generated	253000	0.007194	0.059596	0.017001	0.012336	0.017076	0.024949
Φ-Sat-2	Original	262144	0.0123	0.0362	0.019386	0.0152	0.0192	0.0256
Absolute difference	Generated vs. Φ-Sat-2	9144	0.005106	0.023396	0.002385	0.002864	0.002124	0.000651
Percentage difference (%)	Generated vs. Φ-Sat-2	3.49	41.51	64.63	12.30	18.84	**11.06**	2.54

Table F.5: Reflectance statistics for the G spectral band comparing generated images for the five input textures with the Φ-Sat-2 C3E reference observation at an estimated wind speed of 2 m/s.

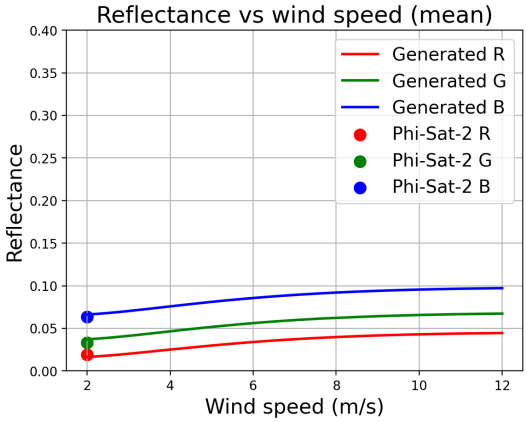
Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	253000	0.020983	0.053600	0.037167	0.027240	0.036954	0.046809
Auckland 2011	Generated	253000	0.027641	0.050303	0.038366	0.032408	0.038291	0.044665
Maui 2015	Generated	253000	0.006006	0.115060	0.037538	0.030736	0.037390	0.050827
Pelagos 2016	Generated	253000	0.019858	0.051621	0.037542	0.033393	0.037503	0.041798
Valdes 2014	Generated	253000	0.006255	0.688790	0.050476	0.012731	0.037708	0.145569
Mean	Generated	253000	0.016149	0.191875	0.040218	0.027302	0.037569	0.065934
Φ-Sat-2	Original	262144	0.0264	0.0457	0.033010	0.0291	0.0329	0.0382
Absolute difference	Generated vs. Φ-Sat-2	9144	0.010251	0.146175	0.007207	0.001798	0.004669	0.027734
Percentage difference (%)	Generated vs. Φ-Sat-2	3.49	38.83	319.86	21.83	6.18	**14.19**	72.60

Table F.6: Reflectance statistics for the B spectral band comparing generated images for the five input textures with the Φ-Sat-2 C3E reference observation at an estimated wind speed of 2 m/s.

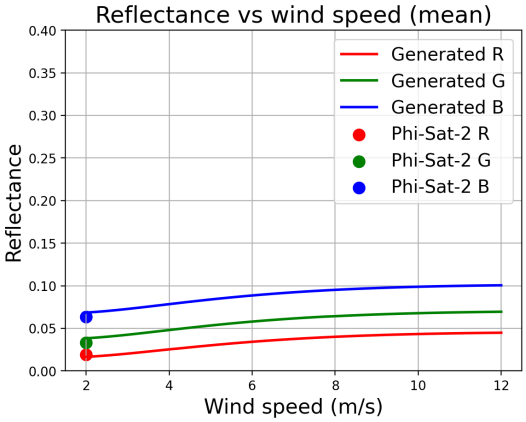
Input Image	Source	n	Min	Max	Mean	p1	p50	p99
Auckland 2006	Generated	253000	0.038211	0.094215	0.066243	0.048902	0.066253	0.083451
Auckland 2011	Generated	253000	0.049708	0.090546	0.068538	0.058206	0.068295	0.079414
Maui 2015	Generated	253000	0.010480	0.202383	0.066575	0.053362	0.066163	0.088616
Pelagos 2016	Generated	253000	0.033282	0.093803	0.068189	0.060978	0.068083	0.075777
Valdes 2014	Generated	253000	0.029806	0.293913	0.071625	0.046199	0.067429	0.119164
Mean	Generated	253000	0.032297	0.154972	0.068234	0.053530	0.067245	0.089284
Φ-Sat-2	Original	262144	0.0574	0.0726	0.063389	0.0601	0.0633	0.0674
Absolute difference	Generated vs. Φ-Sat-2	9144	0.025103	0.082372	0.004845	0.006570	0.003945	0.021884
Percentage difference (%)	Generated vs. Φ-Sat-2	3.49	43.73	113.46	7.64	10.93	**6.23**	32.47

F.2.2. Reflectance Wind Velocity Match

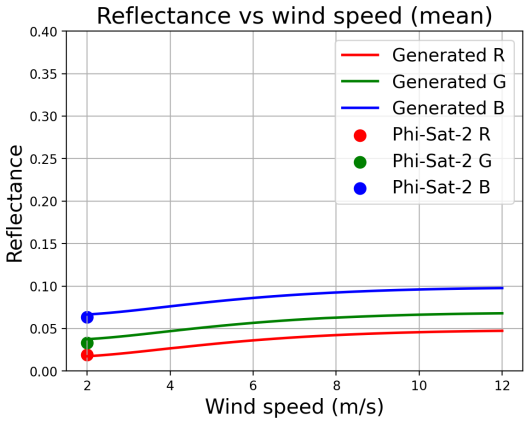
Per Image



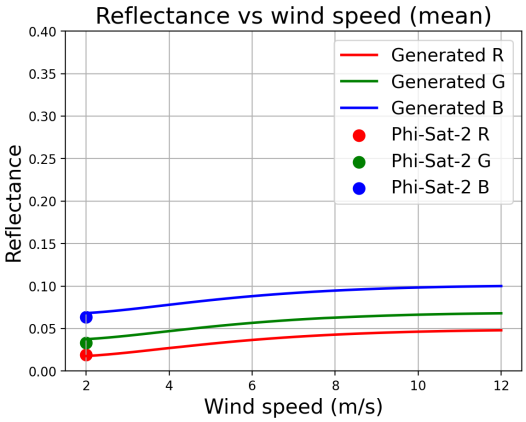
(a) Auckland 2006



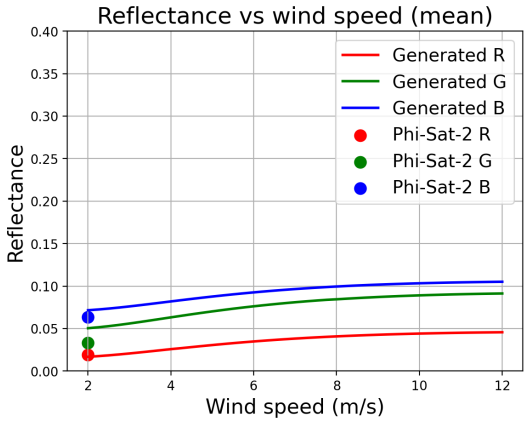
(b) Auckland 2011



(c) Maui 2015



(d) Pelagos 2016



(e) Valdes 2014

Figure F.6: Mean reflectance as a function of wind speed for the C3E observation geometry for five ocean input textures. The curves represent generated reflectance values for the R, G, and B spectral bands, while the markers indicate the true reflectance of the Φ-Sat-2 observation for each band.

Per Band

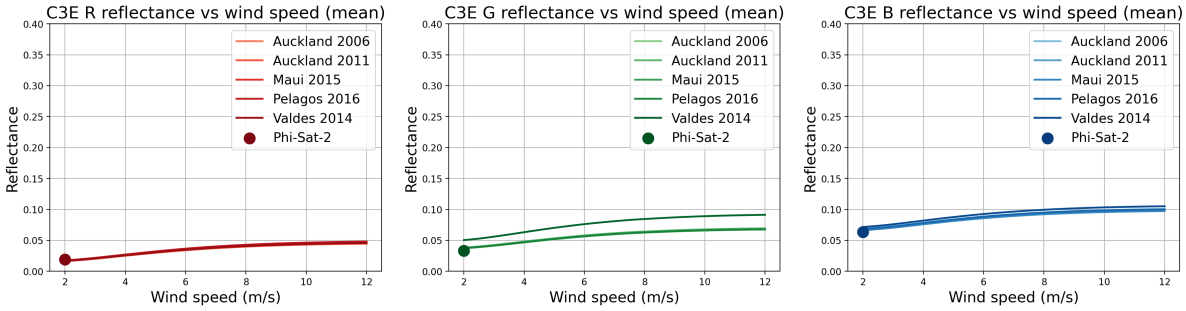


Figure F.7: Mean reflectance as a function of wind speed for the C3E observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

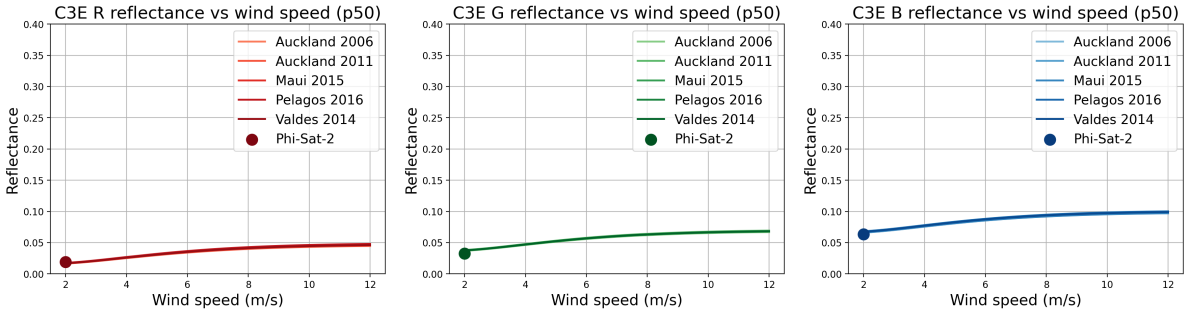


Figure F.8: Median reflectance ($p50$) as a function of wind speed for the C3E observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

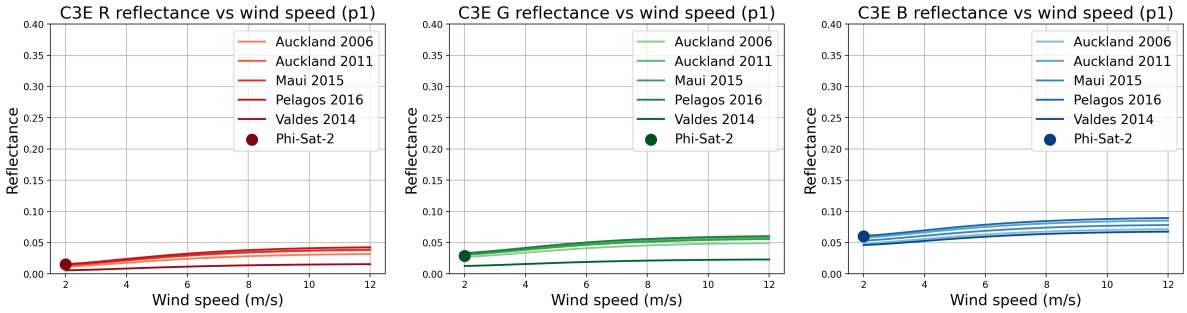


Figure F.9: Lower reflectance percentile ($p1$) as a function of wind speed for the C3E observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

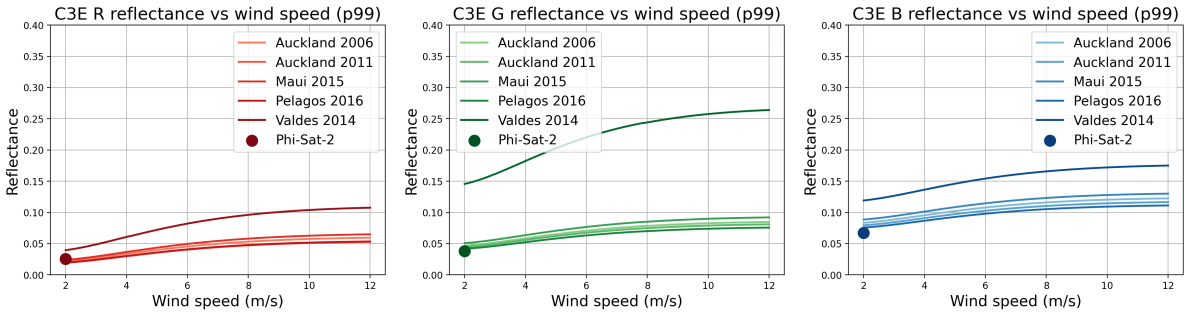


Figure F.10: Upper reflectance percentile ($p99$) as a function of wind speed for the C3E observation geometry, shown separately for the R, G, and B spectral bands. The curves represent generated reflectance values for five different ocean input textures, while the marker indicates the true reflectance of the Φ-Sat-2 observation.

G

Appendix G - Simulation Results

This appendix contains the full benchmark results underlying the summary tables and plots in Chapter 7. The results are reported per simulation run and per target seed for the off-nadir sweep, the time-delay sweep, and the architecture comparison. Its purpose is to provide the complete numerical basis for the averaged trends discussed in the main text.

G.1. Off-nadir Sweep

G.1.1. Pelagos 2016

Table G.1: Benchmark results for the off-nadir sweep in the baseline single Tip-Cue scenario for Pelagos 2016, reported per seed. The time delay is fixed at 5 min and latency L is reported in minutes.

Off-nadir angle [deg]	Target seed	$C_{\text{succ, satellite}}$	$C_{\text{succ, mission}}$	L [min]	V [s]	θ [deg]	AP_{50}	Precision	Recall
5	17	9.5	19.0	5.3	4.7	1.5	0.892	0.864	0.905
5	1	9.0	18.0	5.9	11.4	1.5	0.923	0.750	0.818
5	42	8.0	16.0	5.8	2.6	2.1	0.886	0.696	0.800
10	17	17.0	34.0	5.7	24.0	4.7	0.955	0.919	0.919
10	1	13.5	27.0	6.2	25.5	4.5	0.848	0.771	0.771
10	42	12.0	24.0	6.0	25.1	5.1	0.948	0.828	0.960
15	17	19.5	39.0	5.8	39.3	9.0	0.885	0.830	0.830
15	1	16.0	32.0	6.1	38.5	8.8	0.857	0.744	0.780
15	42	15.5	31.0	6.1	42.1	9.9	0.887	0.816	0.886
20	17	22.0	44.0	5.8	51.8	15.2	0.880	0.800	0.830
20	1	22.0	44.0	5.8	53.3	14.8	0.866	0.830	0.830
20	42	17.5	35.0	5.9	55.8	17.0	0.864	0.833	0.795
25	17	26.5	53.0	5.6	71.1	21.4	0.876	0.815	0.869
25	1	26.5	53.0	5.8	69.9	21.7	0.833	0.746	0.779
25	42	18.5	37.0	5.6	75.2	22.6	0.881	0.804	0.755
30	17	27.5	55.0	5.2	94.2	25.6	0.868	0.846	0.846
30	1	23.5	47.0	5.5	95.6	26.2	0.837	0.825	0.810
30	42	19.0	38.0	5.4	96.8	27.2	0.814	0.760	0.745
35	17	23.0	46.0	5.1	118.6	30.1	0.890	0.821	0.754
35	1	27.5	55.0	5.2	120.4	30.7	0.911	0.821	0.833
35	42	18.5	37.0	5.2	122.3	31.8	0.896	0.771	0.841
40	17	25.0	50.0	5.0	148.3	34.5	0.851	0.833	0.820
40	1	28.0	56.0	4.8	148.4	35.2	0.936	0.875	0.875
40	42	23.0	46.0	4.9	150.0	36.4	0.891	0.852	0.852
45	17	22.5	45.0	4.5	180.5	39.0	0.935	0.804	0.865
45	1	25.5	51.0	4.6	182.9	39.6	0.897	0.850	0.850
45	42	19.0	38.0	4.6	183.2	41.0	0.805	0.760	0.731
50	17	26.5	53.0	4.1	223.0	43.4	0.830	0.815	0.768
50	1	23.0	46.0	4.1	225.9	44.2	0.832	0.780	0.767
50	42	19.0	38.0	4.1	225.1	45.6	0.875	0.826	0.792
55	17	23.5	47.0	3.5	278.4	47.8	0.874	0.904	0.839
55	1	24.5	49.0	3.6	283.7	48.6	0.794	0.766	0.803
55	42	21.0	42.0	3.7	284.1	50.2	0.872	0.894	0.792
60	17	21.5	43.0	3.1	344.8	50.3	0.785	0.811	0.729
60	1	23.0	46.0	3.2	353.4	51.1	0.832	0.807	0.719
60	42	15.5	31.0	3.2	351.7	52.9	0.779	0.775	0.674

G.1.2. Auckland 2006

Table G.2: Benchmark results for the off-nadir sweep in the baseline single Tip-Cue scenario for Auckland 2006, reported per seed. The time delay is fixed at 5 min and latency L is reported in minutes.

Off-nadir angle [deg]	Target seed	$C_{\text{succ, satellite}}$	$C_{\text{succ, mission}}$	L [min]	V [s]	θ [deg]	AP_{50}	Precision	Recall
5	17	7.5	15.0	5.1	3.5	1.5	0.672	0.789	0.714
5	1	5.0	10.0	5.8	11.3	1.5	0.498	0.625	0.476
5	42	7.0	14.0	5.5	1.9	2.1	0.720	0.824	0.667
10	17	12.0	24.0	5.7	24.4	4.7	0.759	0.828	0.686
10	1	9.5	19.0	5.9	26.0	4.5	0.634	0.792	0.594
10	42	7.5	15.0	6.0	25.8	5.1	0.618	0.789	0.600
15	17	14.0	28.0	5.9	38.8	9.0	0.706	0.800	0.636
15	1	14.5	29.0	6.1	38.1	8.8	0.723	0.763	0.725
15	42	9.5	19.0	5.8	41.2	9.9	0.554	0.655	0.559
20	17	18.5	37.0	5.6	52.9	15.2	0.739	0.771	0.661
20	1	15.0	30.0	5.7	55.9	14.8	0.648	0.789	0.577
20	42	14.0	28.0	6.0	49.6	17.0	0.767	0.875	0.667
25	17	20.0	40.0	5.6	71.1	21.4	0.698	0.741	0.645
25	1	21.5	43.0	5.8	73.2	21.7	0.700	0.782	0.662
25	42	15.5	31.0	5.5	75.0	22.6	0.740	0.816	0.689
30	17	22.0	44.0	5.4	95.0	25.6	0.784	0.772	0.688
30	1	20.0	40.0	5.3	95.2	26.2	0.735	0.769	0.667
30	42	12.0	24.0	5.5	97.4	27.2	0.521	0.686	0.490
35	17	23.0	46.0	5.3	118.5	30.1	0.741	0.780	0.719
35	1	17.0	34.0	5.2	119.6	30.7	0.579	0.642	0.557
35	42	14.5	29.0	5.1	121.5	31.8	0.640	0.763	0.617
40	17	20.0	40.0	5.0	149.6	34.5	0.642	0.769	0.625
40	1	19.5	39.0	4.8	148.4	35.2	0.641	0.709	0.629
40	42	16.5	33.0	4.9	149.4	36.4	0.761	0.825	0.688
45	17	20.5	41.0	4.4	179.0	39.0	0.715	0.820	0.641
45	1	17.5	35.0	4.5	182.7	39.6	0.669	0.761	0.614
45	42	12.5	25.0	4.7	183.2	41.0	0.713	0.781	0.568
50	17	17.0	34.0	4.2	225.8	43.4	0.607	0.756	0.540
50	1	19.0	38.0	4.2	225.2	44.2	0.655	0.691	0.585
50	42	14.5	29.0	4.1	225.2	45.6	0.723	0.806	0.630
55	17	18.5	37.0	3.5	274.5	47.8	0.681	0.771	0.607
55	1	18.0	36.0	3.7	284.1	48.6	0.658	0.750	0.581
55	42	12.5	25.0	3.6	283.2	50.2	0.612	0.714	0.532
60	17	16.0	32.0	3.1	350.1	50.3	0.667	0.744	0.542
60	1	14.5	29.0	3.2	353.9	51.1	0.622	0.725	0.509
60	42	12.0	24.0	3.2	352.4	52.9	0.548	0.727	0.490

G.2. Time Delay Sweep

G.2.1. Pelagos 2016

Table G.3: Benchmark results for the time-delay sweep in the baseline single Tip-Cue scenario for Pelagos 2016, reported per seed. The nominal off-nadir angle is fixed at $40°$ and latency L is reported in minutes.

Time delay [min]	Target seed	$C_{\text{succ, satellite}}$	$C_{\text{succ, mission}}$	L [min]	V [s]	θ [deg]	AP_{50}	Precision	Recall
1	17	21.0	42.0	2.9	36.7	18.6	0.785	0.712	0.724
1	1	24.5	49.0	3.0	38.7	20.0	0.865	0.845	0.803
1	42	20.0	40.0	3.0	39.8	19.5	0.869	0.870	0.800
2	17	26.5	53.0	2.8	96.4	13.6	0.856	0.768	0.803
2	1	25.5	51.0	2.9	97.9	13.9	0.847	0.761	0.810
2	42	24.5	49.0	2.9	98.5	14.7	0.935	0.860	0.845
3	17	26.5	53.0	3.1	139.7	31.5	0.861	0.779	0.791
3	1	22.5	45.0	3.1	142.7	32.1	0.860	0.849	0.738
3	42	19.5	39.0	3.1	141.6	32.7	0.874	0.750	0.750
4	17	25.5	51.0	3.9	148.7	34.6	0.877	0.823	0.879
4	1	25.5	51.0	3.9	150.7	35.3	0.877	0.850	0.864
4	42	19.0	38.0	3.7	145.2	35.2	0.882	0.760	0.826
5	17	25.0	50.0	5.0	148.3	34.5	0.851	0.833	0.820
5	1	28.0	56.0	4.8	148.4	35.2	0.936	0.875	0.875
5	42	23.0	46.0	4.9	150.0	36.4	0.891	0.852	0.852
6	17	22.5	45.0	5.8	146.0	34.6	0.778	0.750	0.703
6	1	22.5	45.0	6.0	149.3	35.1	0.812	0.833	0.776
6	42	21.5	43.0	6.0	147.2	34.2	0.822	0.811	0.811
7	17	23.5	47.0	7.0	143.9	34.6	0.849	0.887	0.810
7	1	26.0	52.0	6.9	145.1	34.9	0.837	0.788	0.788
7	42	21.5	43.0	7.0	145.6	34.8	0.904	0.811	0.878
8	17	25.5	51.0	8.1	139.8	34.5	0.859	0.750	0.797
8	1	27.0	54.0	8.1	142.2	34.7	0.810	0.857	0.750
8	42	26.0	52.0	7.8	142.5	35.3	0.891	0.867	0.852
9	17	23.0	46.0	8.7	139.8	34.9	0.825	0.821	0.821
9	1	28.5	57.0	9.2	142.9	34.9	0.896	0.919	0.864
9	42	19.0	38.0	8.8	141.5	35.3	0.869	0.809	0.809
10	17	22.5	45.0	9.7	134.3	35.5	0.883	0.833	0.833
10	1	28.0	56.0	10.1	136.7	34.3	0.873	0.836	0.836
10	42	22.5	45.0	10.1	139.7	35.7	0.870	0.833	0.804

G.2.2. Auckland 2006

Table G.4: Benchmark results for the time-delay sweep in the baseline single Tip-Cue scenario for Auckland 2006, reported per seed. The nominal off-nadir angle is fixed at 40° and latency L is reported in minutes.

Time delay [min]	Seed	$C_{\text{succ, satellite}}$	$C_{\text{succ, mission}}$	L [min]	V [s]	θ [deg]	AP_{50}	Precision	Recall
1	17	14.5	29.0	3.0	42.2	18.6	0.620	0.744	0.558
1	1	22.0	44.0	3.0	39.2	20.0	0.756	0.830	0.710
1	42	12.5	25.0	3.0	42.0	19.5	0.691	0.758	0.610
2	17	16.0	32.0	2.8	96.4	13.6	0.555	0.711	0.542
2	1	18.5	37.0	2.9	97.4	13.9	0.652	0.712	0.617
2	42	17.5	35.0	2.9	97.9	14.7	0.668	0.778	0.648
3	17	18.5	37.0	3.0	137.7	31.5	0.659	0.804	0.578
3	1	17.5	35.0	3.1	142.6	32.1	0.556	0.700	0.583
3	42	13.5	27.0	3.1	144.5	32.7	0.652	0.750	0.675
4	17	18.5	37.0	3.8	149.1	34.6	0.672	0.787	0.578
4	1	18.0	36.0	3.9	151.0	35.3	0.571	0.679	0.571
4	42	13.0	26.0	3.7	146.4	35.2	0.637	0.788	0.520
5	17	20.0	40.0	5.0	149.6	34.5	0.642	0.769	0.625
5	1	19.5	39.0	4.8	148.4	35.2	0.641	0.709	0.629
5	42	16.5	33.0	4.9	149.4	36.4	0.761	0.825	0.688
6	17	18.0	36.0	5.9	146.7	34.6	0.659	0.783	0.571
6	1	16.0	32.0	5.7	148.3	35.1	0.713	0.780	0.571
6	42	12.5	25.0	5.7	143.7	34.2	0.504	0.581	0.532
7	17	21.0	42.0	7.1	145.8	34.6	0.685	0.750	0.689
7	1	20.5	41.0	6.8	144.9	34.9	0.744	0.820	0.745
7	42	17.5	35.0	6.8	141.3	34.8	0.608	0.745	0.636
8	17	13.5	27.0	8.6	138.6	34.5	0.579	0.711	0.519
8	1	21.5	43.0	8.5	142.1	34.7	0.667	0.782	0.606
8	42	17.5	35.0	8.2	142.5	35.3	0.689	0.795	0.648
9	17	16.5	33.0	8.7	139.3	34.9	0.725	0.846	0.623
9	1	19.5	39.0	9.3	140.9	34.9	0.670	0.796	0.542
9	42	14.5	29.0	8.6	142.9	35.3	0.673	0.879	0.580
10	17	16.5	33.0	9.9	136.9	35.5	0.672	0.825	0.559
10	1	18.0	36.0	10.8	136.8	34.3	0.556	0.692	0.507
10	42	17.5	35.0	10.0	137.8	35.7	0.664	0.729	0.614

G.3. Mission Configuration Comparison

Table G.5: Per-run benchmark results for the five baseline configurations in the Tip-and-Cue scenario using the `reflection_offnadir_glint_255` results. Latency L is reported in minutes.

Configuration	Target seed	$N_{\text{satellites}}$	$C_{\text{succ, satellite}}$	$C_{\text{succ, mission}}$	L [min]	V [s]	θ [deg]	$AP50$	Precision	Recall
Single Tip-Cue system	1	2	28.0	56.0	4.8	148.4	35.2	0.936	0.875	0.875
Single Tip-Cue system	17	2	25.0	50.0	5.0	148.3	34.5	0.851	0.833	0.820
Single Tip-Cue system	42	2	23.0	46.0	4.9	150.0	36.4	0.891	0.852	0.852
8x2 VHR constellation	1	16	2.4	39.0	1.0	79.8	0.0	0.867	0.750	0.867
8x2 VHR constellation	17	16	3.1	49.0	1.0	79.6	0.0	0.826	0.766	0.875
8x2 VHR constellation	42	16	2.8	44.0	1.0	80.2	0.0	0.920	0.800	0.936
4x4 VHR constellation	1	16	2.6	42.0	1.0	79.7	0.0	0.899	0.750	0.857
4x4 VHR constellation	17	16	2.6	42.0	1.0	79.9	0.0	0.940	0.778	0.955
4x4 VHR constellation	42	16	2.9	46.0	1.0	80.0	0.0	0.839	0.767	0.868
4x2x2 Tip-Cue constellation	1	16	20.9	335.0	4.9	151.5	31.4	0.835	0.790	0.783
4x2x2 Tip-Cue constellation	17	16	20.4	327.0	5.0	152.9	32.5	0.842	0.775	0.782
4x2x2 Tip-Cue constellation	42	16	20.4	326.0	5.0	152.2	31.1	0.820	0.778	0.797
12 independent satellites	1	12	14.6	175.0	53.5	98.4	21.3	0.905	0.841	0.850
12 independent satellites	17	12	14.0	168.0	52.9	100.3	20.9	0.882	0.800	0.857
12 independent satellites	42	12	13.3	160.0	69.3	99.5	20.3	0.888	0.833	0.879

Table G.6: Per-run task-flow results for the five baseline configurations in the Tip-and-Cue scenario using the `reflection_offnadir_glint_255` results. The first four columns are taken from the raw `Overview` sheets in the uploaded `results_*.xlsx` files. The final column matches the corrected benchmark-overview value of $C_{\text{succ, mission}}$ for the same run.

Configuration	Target seed	Tasks, Tip observed	Tasks, Tip sent	Tasks, Cue started	Tasks, Cue observed	Tasks, Cue confirmed, GT
Single Tip-Cue system	1	79	70	70	70	56
Single Tip-Cue system	17	76	68	68	70	50
Single Tip-Cue system	42	59	54	54	54	46
8x2 VHR constellation	1	-	-	-	48	39
8x2 VHR constellation	17	-	-	-	59	49
8x2 VHR constellation	42	-	-	-	52	44
4x4 VHR constellation	1	-	-	-	56	42
4x4 VHR constellation	17	-	-	-	54	42
4x4 VHR constellation	42	-	-	-	52	46
4x2x2 Tip-Cue constellation	1	601	483	480	482	335
4x2x2 Tip-Cue constellation	17	569	466	465	465	327
4x2x2 Tip-Cue constellation	42	564	451	451	460	326
12 independent satellites	1	386	292	531	229	175
12 independent satellites	17	390	302	573	229	168
12 independent satellites	42	380	290	631	212	160

H

Appendix H - International Astronautical Congress Paper

This appendix contains the paper presented during an oral session at the International Astronautical Congress (IAC) in Sydney in October 2025. The content reflects the state of the project at that time and is therefore not fully aligned with the experiments and results presented in this thesis.

Following completion of the thesis, the work will be revised and extended based on the final results. A refined version will then be prepared as a white paper with subsequent journal submission.

76th International Astronautical Congress (IAC 2025), Sydney, Australia, 29 Sep-3 Oct 2025.

IAC-25,B1,8,5,x95270

Laying the Foundation for AI-Based Tip and Cue: Enabling High Spatial and Temporal Resolution in Remote Sensing

Nadine A. Duursma

Department of Robotics and Aerospace Engineering, Delft University of Technology, Mekelweg 5, 2628 CD Delft, The Netherlands / European Space Agency Φ-lab, Via Galileo Galilei, 1, 00044 Frascati, Italy

Abstract

Current Earth Observation (EO) satellites face trade-offs between field-of-view, spatial resolution, and revisit time, limiting their ability to rapidly detect and respond to events such as wildfires, maritime threats, and natural disasters. The AI-based Tip and Cue strategy addresses these limitations by pairing a wide field-of-view "Tip" satellite for anomaly detection with a high-resolution "Cue" satellite for targeted imaging.

This work presents the first integrated framework for AI-based Tip and Cue that integrates two complementary components: (1) a mission simulation environment that models orbital dynamics, constellation architectures, sensor pointing, and tasking logic under realistic orbital conditions; and (2) a dataset generation pipeline that, based on the simulated orbital positions, produces physically realistic synthetic imagery including off-nadir and radiometric effects for benchmarking onboard AI detection models.

On the mission side, whale targets are propagated as mobile maritime objects with realistic headings, speeds, and land avoidance. A Walker-Delta constellation generator, dynamics-based sensor stabilization, and autonomous tasking logic were implemented to quantify coverage, latency, and off-nadir pointing demands. On the dataset side, the off-nadir pointing requests from the simulation are combined with bidirectional reflectance distribution functions (BRDF) and atmospheric irradiance models in a physics-based rendering pipeline to simulate geometric distortion and radiometric effects such as sun glint, generating training and evaluation data for onboard AI.

A 24-hour case study demonstrates the framework's ability to compare mission architectures: a 16-satellite Walker-Delta constellation and a two-satellite Tip and Cue configuration performed 52 and 54 whale detections respectively, with Tip and Cue achieving this at the cost of higher latency and off-nadir angles but with an order of magnitude fewer satellites.

By combining system-level mission performance with AI detection realism, this work aims to lay a foundation for robust evaluation of AI-based Tip and Cue missions, supporting the design of Earth observation missions with improved responsiveness, coverage, and detection accuracy.

Keywords: Tip and Cue, Off-Nadir Imaging, Onboard AI, Mission Simulation, Earth Observation, Whale Detection.

1. Introduction

Earth observation (EO) satellites face trade-offs between spatial resolution, swath width, and revisit time [1]. Wide-swath systems, like Sentinel-2, can capture large areas frequently but lack the spatial detail needed to identify small or distant targets. High-resolution systems, such as WorldView-3, can resolve fine details but cover much smaller areas and revisit less often. These constraints mean that, in practice, no single satellite can simultaneously provide high spatial and temporal resolution with broad coverage.

The Tip and Cue strategy offers a way around these limitations. In this approach, a wide field-of-view "Tip" satellite surveys large areas to detect events or anomalies, while a high-resolution "Cue" satellite is tasked to capture detailed imagery of the same location shortly afterwards [2]. This concept has the potential to be applied in contexts such as wildfire monitoring, maritime surveillance, and disaster response. Traditionally, the Tip and Cue principle relies on transmitting Tip satellite detections to the ground for processing, and then uplinking a tasking command to a Cue satellite operating independently. While operationally proven, this process introduces latency, often hours, reducing the value of the cueing action for rapidly evolving events [3].

Integrating onboard artificial intelligence (AI) into the satellites enables detection and tasking decisions to be made directly in orbit, reducing latency to minutes or even seconds. This can be transformative for time-sensitive scenarios: a wildfire front can be detected before rapid expansion, a vessel engaged in illegal fishing can be documented before it leaves the scene, or an oil spill can be mapped before it disperses beyond containment.

However, AI-based Tip and Cue concepts studied so far typically assume idealised nadir viewing conditions. In these studies, AI model performance, for example in wildfire [2] or methane [4] detection, is evaluated without accounting for off-nadir pointing, realistic orbital dynamics, or the time delays between Tip detection and Cue acquisition. In reality, many

observations, especially urgent cues, occur at off-nadir angles, where the sensor is pointed away from directly beneath the satellite. Such imaging suffers from geometric degradation in the form of increased ground sampling distance and perspective changes, as well as radiometric degradation due to altered illumination geometry, longer atmospheric paths, and sun glint over reflective surfaces like water. These effects can reduce the accuracy of onboard AI detections, yet they are rarely accounted for in mission design or performance modelling.

This work addresses that gap by developing the first integrated framework for AI-based Tip and Cue missions, linking two complementary components:

1. Mission simulation environment: modelling constellation architectures, orbital dynamics, whale target propagation, tasking logic, and sensor pointing and stabilization. This enables benchmarking of coverage, latency, tasked targets, and off-nadir pointing statistics under realistic mission conditions.
2. Dataset generation and AI pipeline: using simulated orbital states, pointing angles, and solar incidence from the mission simulation to produce physically realistic synthetic imagery with off-nadir and radiometric effects. This ensures that AI models are trained and evaluated under the same geometric and illumination conditions encountered in orbit.

The framework is applied to a case study on whale detection, chosen for its combination of high-resolution requirements and uncertain maritime occurrence. Whales are propagated as mobile maritime targets, detected by a Tip satellite, and cued to a high-resolution Cue satellite. Synthetic off-nadir and sun glint imagery are generated based on satellite and target position and provide training and evaluation data for AI-based detection. Together, these components enable systematic benchmarking of detection accuracy, coverage, latency, and observation efficiency, and allow for comparison between classical constellations and Tip and Cue architectures.

2. Background

The Tip and Cue concept involves a two-satellite formation in which a wide field-of-view *Tip* satellite surveys large areas to detect anomalies and a high-resolution *Cue* satellite captures detailed imagery of detected features of interest, as illustrated in Fig. 1. Traditionally, Tip and Cue implementations depend on ground-based processing and human tasking, which can introduce delays that limit responsiveness to fast-changing events [5]. Integrating onboard AI enables autonomous detection and cueing directly in orbit, significantly reducing latency, while achieving wide-area coverage with a high spatial resolution.

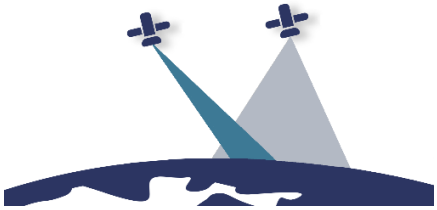


Fig. 1. The Tip and Cue strategy with a wide field-of-view satellite (Tip) on the right, followed by a complementary high spatial resolution sensor (Cue) on the left.

The concept of AI-based Tip and Cue remains only marginally explored in open literature. Prior research often addresses individual components, like on-board AI performance evaluation or communication frameworks [6], without fully integrating these into an autonomous, multi-satellite operational framework. Commercial demonstrations have shown feasibility, but typically rely on human intervention [5] and do not provide open datasets or reproducible benchmarks.

A defining operational feature of Cue satellites is their ability to point off-nadir, enabling imaging of targets outside the sub-satellite track and expanding opportunities for follow-up. However, this introduces performance penalties in two categories:

- Geometric effects: increased ground sampling distance (GSD), target distortion, and perspective changes.
- Radiometric effects: illumination geometry changes, increased atmospheric path length, and sun glint over water, as shown in Fig. 2.

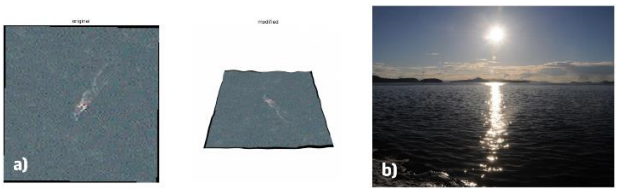


Fig. 2. Simulated off-nadir effects showing (a) a change in viewing perspective and increased ground sampling distance, and (b) sun glint forming a glitter path on the ocean. Satellite image © 2022 Maxar Technologies.

These effects degrade image quality and can impact onboard AI detection accuracy, yet they are rarely incorporated into current Tip and Cue mission modelling.

The literature review identified three research gaps:

1. Mission simulation gap: Absence of an open, end-to-end simulation framework coupling orbital tasking logic, physics-based off-nadir rendering, and AI inference in a unified workflow.

2. Dataset and AI gap: Lack of systematic evaluation of geometric and radiometric effects on AI-based detection performance, due to missing datasets that capture these effects.
3. Benchmarking gap: no established methodology to compare AI-based Tip and Cue configurations against classical constellations across orbital, sensor, and processing architectures.

As part of this work, over fifty potential AI-based Tip and Cue applications were identified through literature and expert consultation. Based on imaging characteristics, impact, novelty, and data availability, wildlife monitoring, and particularly whale detection, was selected as a demonstration use-case. This application presents detection challenges similar to those in other maritime domains, distinguishing small or partially visible objects from ocean backgrounds, and can therefore be extended to vessel, oil spill, iceberg, or marine litter detection in future work.

In summary, the current state of the art provides a conceptual basis for AI-based Tip and Cue but lacks integrated simulation tools, standardized performance benchmarks, and evaluation under realistic off-nadir conditions. The work presented here directly addresses these gaps.

3. Methods

3.1 Mission Simulation Component

The mission simulation environment models the complete Tip and Cue chain, from the movement of maritime targets to satellite tasking and stabilized sensor pointing. It is designed to reproduce realistic orbital conditions and operational constraints, while allowing for different constellations, sensors, and tasking policies.

3.1.1 Target Propagation

Whale targets are generated as synthetic proxies for small, mobile anomalies in the maritime domain, as shown in Fig. 3. Initial positions are sampled uniformly over the oceans, using a high-resolution GSHHG land mask to exclude land areas. Each whale follows a stochastic motion model: headings evolve with Gaussian diffusion, while speeds vary between 0.5-3.0 m/s according to an Ornstein-Uhlenbeck process. A land-avoidance routine ensures that trajectories remain ocean-bound. The resulting target set provides a dynamic, realistic background against which detection and tasking can be evaluated.

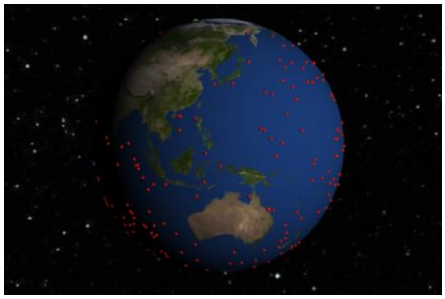


Fig. 3. Synthetic whale targets (red) are sampled uniformly over ocean regions using a high-resolution land mask, with subsequent motion governed by a stochastic heading and speed model and constrained to remain ocean-bound.

3.1.2 Constellation Setup

Satellite constellations are instantiated using Orekit's Walker-Delta generator, which distributes satellites across orbital planes according to the total number of satellites T, planes P, and phasing factor F. A helper function ensures that F satisfies the Walker coprimality condition for valid slotting. This enables systematic comparison of large very-high resolution (VHR) constellations (e.g., 16 satellites across 8 planes) with minimal Tip and Cue pairs (one Tip and one Cue), while maintaining consistent orbital mechanics. Fig. 4. illustrates a Walker-Delta constellation with four planes and four Tip-Cue pairs per plane as an example configuration.

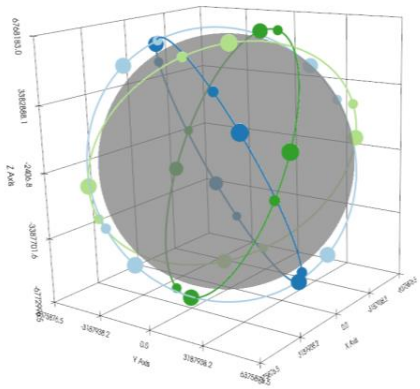


Fig. 4. A Tip and Cue Walker-Delta constellation of four planes, four T/C satellite-pairs per plane, at 60-degree inclination.

3.1.3 Orbital Propagation

The orbital elements are propagated to determine the position of both satellites over time using Orekit and PASEOS [7]. Tip satellites are nadir-pointed and scan wide areas, where the nominal Tip field-of-view

propagation, without tasking or stabilization effects, is visualized in Fig. 5.

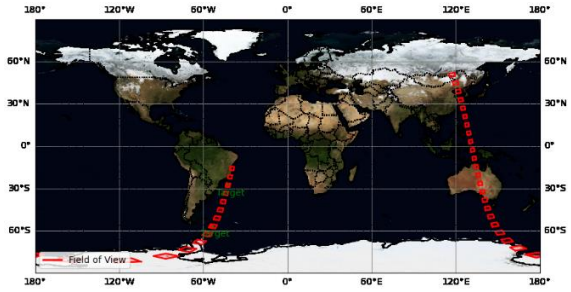


Fig. 5. Visualization of the orbital propagation of field-of-view windows from the Tip satellite, where the Cue satellite is tasked after the target coincides the window.

3.1.4 Tasking Logic and Illumination Constraints.

A detection occurs when a whale target falls within the Tip footprint and is illuminated by the Sun. To model Tip's onboard AI accuracy, a true positive rate of 0.85 was applied, so not all observations are passed to Cue, with a confirmation delay of 90 seconds. Positively confirmed targets are assigned to Cue satellites by propagating forward to the expected tasking delay and minimizing the satellite-target distance at that time. Tasking also requires the Cue satellite to be illuminated and the target to remain in daylight. Cue satellites maintain a task queue, with the next observation selected using a pointing cost metric that minimizes angular retargeting. Fig. 6 shows a simulation in which two targets are observed sequentially.

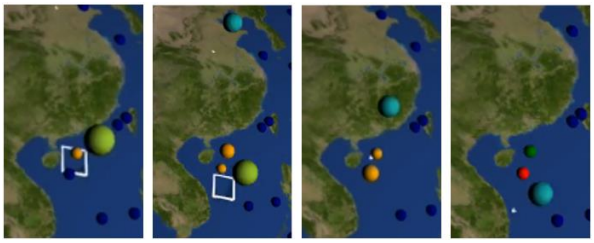


Fig 6. Simulation of Tip and Cue tasking logic, showing how a target is observed by the Tip satellite (orange), tasked (big orange), observed by the Cue satellite (small orange), and confirmed (true positive = green, false negative = red), with illumination and field-of-view conditions satisfied.

3.1.5 Sensor Pointing and Stabilization

After a Cue satellite is tasked, the imaging sensor must be reoriented from its current orientation to the commanded line-of-sight. This maneuver is divided into two phases: (i) a slew maneuver to rotate the spacecraft under actuator constraints, and (ii) a stabilization phase to suppress residual oscillations before imaging.

Slew Maneuver

The retargeting profile is constrained by the maximum angular velocity $\omega_{\max}$ and angular acceleration $\alpha_{\max}$. As a result, the slewing maneuver is divided into three distinct phases: an acceleration phase, where angular velocity increases linearly at rate $\alpha_{\max}$ until it reaches $\omega_{\max}$; a constant-rate phase, where the rotation proceeds at $\omega_{\max}$; and a deceleration phase, where angular velocity decreases linearly back to zero.

For small angular separations Δ, the maximum velocity is never reached, and the profile simplifies to a triangular shape, consisting only of acceleration followed by immediate deceleration, for which the acceleration and total slewing times are given by Eq. 1. and 2. respectively.

$$t_{\text{acc}} = t_{\text{dec}} = \sqrt{\frac{\Delta}{\alpha_{\max}}} \tag{1}$$

$$t_{\text{slew}} = t_{\text{acc}} + t_{\text{dec}} = 2\sqrt{\frac{\Delta}{\alpha_{\max}}} \tag{2}$$

For larger separations, the maneuver adopts a trapezoidal profile, the constant rate phase at $\omega_{\max}$ is reached within the slewing times given by Eq. 3. and 4.

$$t_{\text{acc}} = t_{\text{dec}} = \frac{\omega_{\max}}{\alpha_{\max}} \qquad t_{\text{const}} = \frac{\Delta - \omega_{\max}^2/\alpha_{\max}}{\omega_{\max}} \tag{3}$$

$$t_{\text{slew}} = t_{\text{acc}} + t_{\text{const}} + t_{\text{dec}} \tag{4}$$

Stabilization Phase

Following the slew, the spacecraft must dwell until pointing disturbances decay. In this work, stabilization is consistently modeled as a second-order damped response characterized by natural frequency ω_n and damping ratio ζ. The 2% settling time is approximated by Eq. 5.

$$t_{\text{settle}} \approx \frac{4}{\zeta\,\omega_n} \tag{5}$$

To represent operational behavior, this duration is scaled with commanded angular displacement, such that large slews require the full settling time while smaller retargetings converge more rapidly. Specifically, the stabilization time is reduced proportionally for angles below 45°, but cannot exceed the base value for larger slews. The total time required for a retargeting maneuver is given by Eq. 6.

$$t_{\text{point}} = t_{\text{slew}} + t_{\text{settle}} \tag{6}$$

Sensor Trajectory

Once the target orientation is defined, the time history of the maneuver is constructed by analytically integrating the angular kinematics across the three phases of motion, acceleration, constant-rate rotation, and deceleration.

- During the acceleration phase, the angular velocity increases linearly with time, and the angular displacement grows quadratically, following the relation $\theta(t) = 1/2\,\alpha_{\max} t^2$.
- In the constant-rate phase, the angular velocity remains fixed at $\omega_{\max}$, and the angular displacement increases linearly as $\theta(t) = \omega_{\max} t + \theta_0$.
- During deceleration, the profile mirrors the acceleration phase in reverse: angular velocity decreases linearly, and angular displacement again follows a quadratic curve until the target angle is reached with zero terminal rate.
- During the stabilization phase, the trajectory is held fixed at the target Euler angles, but the angular velocity and acceleration components are modelled at non-zero to prevent the observation call from being executed before full stabilization has been achieved.

This models a continuous trajectory describing the attitude angle, angular rate, and angular acceleration at each timestep.

Because the Cue satellite is typically tasked several minutes before reaching the target, it can begin reorienting well in advance. As a result, stabilization can occur progressively during its approach, allowing the sensor to be settled and observation-ready by the time the target enters the field of view, as illustrated in Fig. 7.

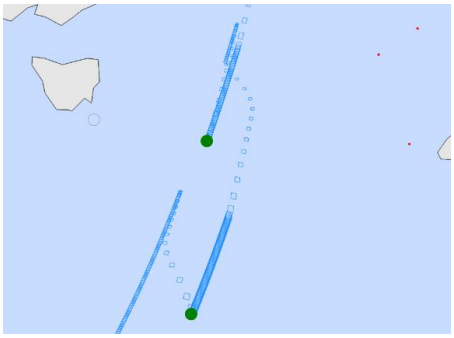


Fig. 7. Field-of-view ground trace of the Cue satellite during tasking of two targets. After completing the first observation, the satellite slews back to nadir before executing the second task. Following the second acquisition, it returns to nadir once more, illustrating the retargeting and recovery behavior between observations.

3.1.6 Performance Metrics

At each timestep, the simulation logs specific events:

- Number of targets observed and confirmed by Tip.
- Number of tasked targets observed and confirmed by Cue.
- Latency between Tip observation and Cue observation.
- Viewing time left after the first observation.
- Off-nadir angles and derived ground sampling distance (GSD).

These outputs enable systematic benchmarking of coverage, latency, and observation efficiency for alternative Tip and Cue configurations.

3.2 Dataset Generation and Onboard AI Component

The dataset generation and onboard AI component builds on the mission simulation framework by producing conditions computed in Section 3.1. This ensures that the same off-nadir angles, delays, and lighting constraints driving the Tip and Cue mission simulation are used to train and evaluate onboard AI models.

3.2.1 Orbital Inputs

Satellite position, velocity, and pointing geometry are imported directly from the mission simulation environment. This provides geodetic target coordinates, off-nadir angles, and sensor footprints for any timestep, ensuring that rendering is tied to actual orbital dynamics rather than abstract scenarios.

The baseline configuration is derived from Sentinel-2-like Tip satellites (SSO, 786 km, 290 km swath, 10 m GSD) and WorldView-3-like Cue satellites (SSO, 617 km, 13.1 km swath, 0.31 m GSD). For each Cue tasking event, the simulation supplies the satellite pose and target state, which define the rendering geometry.

3.2.2 Rendering Pipeline

Off-nadir imaging effects are modelled using a physically based rendering approach. The workflow consists of:

1. Scene Geometry: Geodetic coordinates of the target are converted to Earth-Centered Earth-Fixed coordinates, and the camera pose is set according to the required off-nadir angle.
2. Surface Modelling: Synthetic ocean Digital Elevation Models (DEMs) are generated in Python to represent wave patterns with heights from 0.05-0.5 m, as shown in Fig. 8.

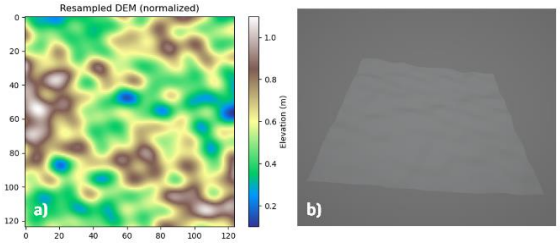


Fig. 8. Synthetic ocean DEM generated by a combination of sine and cosine functions, with a) amplitude profile and b) 3D mesh export.

3. Radiometric Effects: Radiance is computed with a ray-tracing renderer (Mitsuba [8]) using water-surface bidirectional reflectance distribution function (BRDF) parameters, combined with spectral irradiance from the SMARTS radiative transfer model [9]. This captures sun glint, atmospheric path length effects, and illumination geometry changes.
4. Spectral to DN Conversion: Band-specific radiance is integrated over sensor spectral response curves, then converted into 8-bit digital numbers suitable for AI model input.

The output is shown in Fig. 9. This process produces realistic synthetic images at various off-nadir angles, allowing controlled experiments on AI sensitivity to geometric and radiometric distortions.

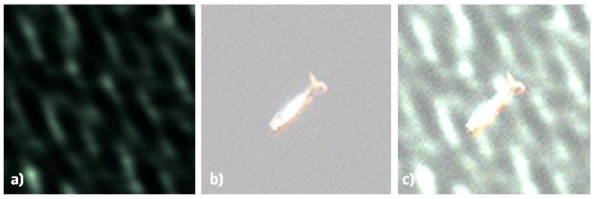


Fig. 9. Artificially generated sun glint image showing a) computed solar radiance component, b) base image, and c) combined result. Satellite image © 2022 Maxar Technologies.

3.2.3 AI Model Integration

To complete the framework in on-going work, the detection stage will use a DEIM-DETR transformer-based object detector [10], fine-tuned on synthetic whale images rendered at operational off-nadir angle ranges. Whale patches are derived from the 633-image Whales from Space dataset [11], cropped to avoid consistently centered targets, and augmented with radiometric effects from the rendering pipeline. Training follows COCO format, with augmentations for variable lighting and perspectives.

After training the model, several experiments are planned to be conducted:

- Hyperparameter setting: The effects of model size and hyperparameter setting on detection performance.
- Off-nadir angle limits: Determining the maximum off-nadir angle that maintains acceptable detection accuracy.
- Radiance modelling effects: Quantifying the impact of modelling radiance effects (sun glints) on detection performance.

3.2.4 Benchmark Definition

To link the dataset pipeline with mission performance, a benchmark will be defined across both AI and mission metrics:

- AI-level: mean average precision (mAP), precision, recall, F1-score.
- Mission-level: coverage (number of unique AOIs cued), latency (time between Tip and Cue observation), viewing time left (time between first Cue observation until out of nadir limits), and observation efficiency (satellites utilized).

Performance can then be expressed in two ways:

1. Detection throughput: the number of valid events (Tip detection - Cue observation - AI detection) successfully handled per satellite per day.
2. Constellation efficiency: the number of satellites saved relative to a Walker constellation baseline, while maintaining threshold detection performance.

4. Results

4.1 Mission Simulation Scenario

The mission simulation was applied to compare two mission architectures:

- Large VHR Walker-Delta constellation: 16 satellites distributed in 8 orbital planes (2 satellites per plane), each with Cue-like imaging parameters (SSO, 617 km, 13.1km swath).
- Minimal AI-based Tip and Cue pair: one wide-swath Tip satellite (SSO, 617 km, 290km swath) and one very-high-resolution Cue satellite with 5 minute spacing-delay (13.1km swath).

Both scenarios were run over a 24-hour simulation horizon with identical target initialization (500 whales). The AI-based Tip and Cue configuration was evaluated with maximum off-nadir limits of both 30°, 40° and 50°. The main performance indicators are summarized below.

4.1.1 Detection and Tasking Performance

The constellation observed 52 distinct whale targets within the 24-hour simulation period, and the AI-based Tip and Cue system observed 59 by Tip, after which 54 were correctly classified as whales, thus passed on and observed by Cue. This demonstrates that a minimal Tip and Cue pair can, under favorable orbital conditions, closely match, or even slightly beat the coverage of a large Walker-Delta constellation. The difference lies in efficiency: the Walker-Delta required 16 spacecraft to achieve this performance, while the Tip and Cue pair achieved the same number of observations with only two satellites.

4.1.2 Latency

In the Walker-Delta constellation, Cue satellites observed targets immediately upon visibility, resulting in no additional tasking delay. In contrast, the Tip and Cue pair exhibited longer tasking delays, with median latencies of 248 seconds at 30°, 222 seconds at 40° and

184 seconds at 50° maximum off-nadir from first observation, as shown in Fig. 10., 11., and 12. respectively. This occurs because the Cue satellite must wait longer to observe the target at 30° than at 40° or 50°.

These delays can be reduced by permitting higher off-nadir acquisitions, enabling earlier look-ahead observations at the cost of degraded ground sampling distance, thus stronger geometric and radiometric distortions.

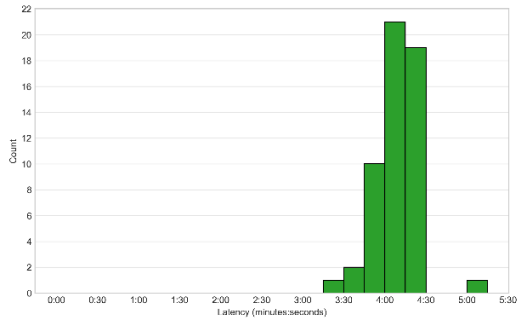


Fig. 10. Latency distribution at 30° maximum off-nadir angle requirement, with a median of 248 s or 4:08 min.

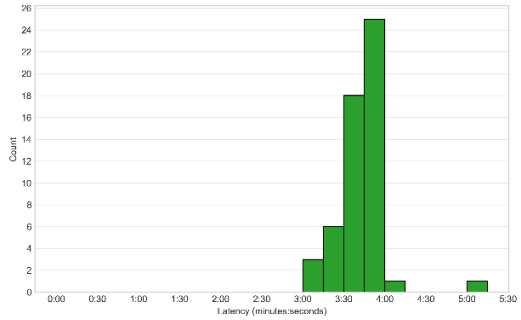


Fig. 11. Latency distribution at 40° maximum off-nadir angle requirement, with a median of 222 s or 3:42 min.

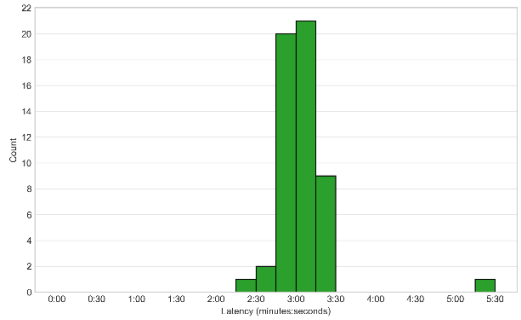


Fig. 12. Latency distribution at 50° maximum off-nadir angle requirement, with a median of 184 seconds or 3:04 min.

4.1.3 Off-nadir Pointing

Cue satellites in the Walker-Delta constellation observed whales at nadir, as the tasked targets were evaluated directly when in view. On the other hand, the minimal Tip and Cue configuration required observed the targets at 30°, 40° and 50°, depending on the imposed off-nadir limit. This keeps the ground sampling distance at 0.41 m, 0.52 m, and 0.75 m for 30°, 40° and 50° respectively. Allowing longer tasking latencies would reduce required off-nadir angles, highlighting the trade-off between image quality and timeliness.

4.1.4 Viewing Time

The duration for which a Cue satellite can continue observing a target after the first acquisition depends strongly on the permitted off-nadir angle. At tighter limits (30°), viewing opportunities are short, with a median of 95 seconds available before the target leaves the field of regard. Increasing the maximum off-nadir angle to 40° extends the available viewing time to 146 seconds, while 50° more than doubles it to 221 seconds on average (Fig. 13-15). In contrast, a Walker-Delta constellation observing targets directly at nadir achieves half this duration, as each satellite propagates from nadir to the off-nadir limit without the extended look-ahead possible in the Tip and Cue system.

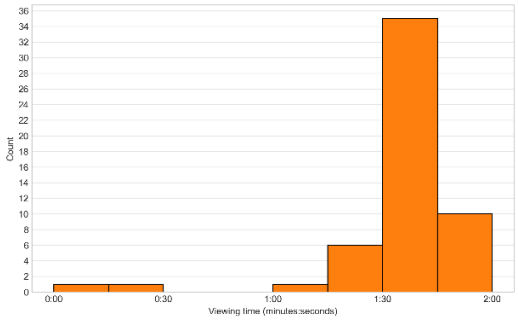


Fig. 13. Viewing time distribution after the first Cue observation, at 30° maximum off-nadir angle requirement, with a median of 95 s or 1:35 min.

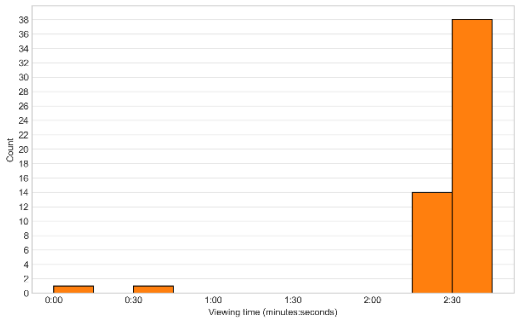


Fig. 14. Viewing time distribution after the first Cue observation, at 40° maximum off-nadir angle requirement, with a median of 146 s or 2:26 min.

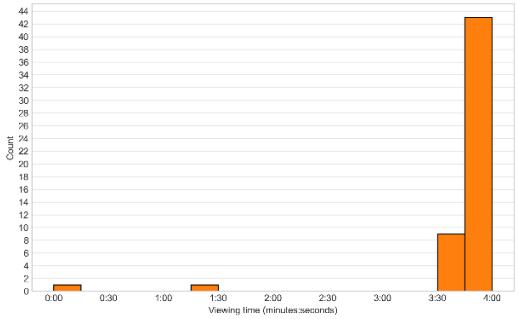


Fig. 15. Viewing time distribution after the first Cue observation, at 50° maximum off-nadir angle requirement, with a median of 221 s or 3:41 min.

4.1.5 Summary

The results show that a minimal Tip–Cue pair can achieve comparable target coverage to a large Walker-Delta constellation with an order of magnitude fewer satellites. This efficiency, however, is offset by longer tasking latencies and the need for frequent off-nadir slews. Allowing larger off-nadir angles reduces latency and extends viewing time, but at the cost of degraded ground sampling distance and stronger geometric and radiometric distortions. These trade-offs underscore that mission performance cannot be assessed on coverage alone, but must also account for the quality of the imagery acquired.

Ongoing simulations will evaluate how Tip and Cue performance responds to variations in cueing delay, orbital altitude, sensor design, and constellation configuration. To capture the consequences of these conditions on detection reliability, Section 4.2 extends the analysis by examining how off-nadir acquisitions affect imaging quality.

4.2 Dataset Effects

The second component of the framework uses the orbital states, pointing angles, and illumination conditions from Section 4.1 to generate synthetic imagery for AI training and evaluation. This ensures that the same degradations driving mission performance also shape dataset realism.

4.2.1 Off-nadir Imaging Demonstration

Fig. 16. presents example maritime renderings at fixed off-nadir angles under varying solar incidence conditions by propagating time, generated with the physics-based rendering pipeline. These visualizations capture two types of changes relative to nadir imagery:

- Geometric effects: an increase in ground sampling distance and visible target elongation with larger off-nadir angles.
- Radiometric effects: brightness and contrast variation due to changing solar geometry, atmospheric path length differences, and sun glint patterns over the ocean surface.

The renderings confirm that the pipeline is capable of simulating the conditions most likely to challenge onboard AI-based detection, from which actual model training and evaluation can follow.

Fig. 16. Simulated off-nadir renderings of the same target at varying solar incidence angles. Cue satellite and target positions are held constant, isolating the effect of illumination geometry. Satellite image © 2022 Maxar Technologies.

4.2.2 Ongoing Work

At this stage, the rendering pipeline is being validated by comparing spectral radiance values and glint patterns against Φsat-2 imagery. Ongoing work focuses on training AI models on the synthetic datasets and incorporating their evaluation into the simulation framework, so that mission comparisons capture coverage and latency alongside the effect of off-nadir conditions on detection throughput and constellation efficiency.

5. Discussion

5.1 Principal Findings

The whale detection case study demonstrates that a minimal Tip and Cue pair can achieve the same number of detections within 24 hours as a Walker-Delta constellation an order of magnitude larger. This result highlights the potential efficiency of AI-based Tip and Cue, but only when higher latency and frequent off-nadir slews are accepted. These slews alter the acquired imagery by increasing ground sampling distance, distorting target shape, and introducing radiometric effects such as sun glint. The altered data reduce the reliability of onboard AI detection, underscoring that mission design cannot be assessed on coverage alone. Latency, viewing geometry, and data quality together determine whether timely and accurate detections are achievable.

5.2 Assumptions and Limitations

The current results provide an indication of Tip and Cue mission feasibility and effectiveness, but several simplifying assumptions limit the strength of the conclusions.

From a detection perspective, the Tip satellite is assumed to provide tasking at a fixed true positive rate:

therefore, approximately 85% of whales at 10 m GSD are correctly flagged and passed on to Cue. In practice, Tip confirmations will suffer from false positives and false negatives depending on sea state, varying backgrounds, or partial visibility. The onboard AI stage of the Tip satellite is not modelled in the present study. Due to the lack of suitable training data, this limitation will not be addressed in ongoing work, but future studies should more accurately quantify the implications of imperfect Tip detections for cueing loads and missed opportunities.

The whale target model also neglects resurfacing cycles and occlusion by environmental effects such as whitecaps, haze, or clouds. This likely leads to an overestimate of detections and underestimates the impact of longer latencies. Clouds are particularly important for maritime imaging, as they can obscure both Tip and Cue detections; their absence is a significant simplification.

At the mission level, the Walker constellation and the Tip-Cue pair are modeled with the same orbital and stabilization parameters, so pointing dynamics are treated consistently. The main difference lies in architecture. A Walker constellation provides redundancy, parallel coverage, and robustness to single-satellite failures, while distributing downlink and tasking load across many spacecraft. A minimal Tip-Cue pair is far more efficient in satellite count and cost, but inherently less resilient: performance depends on the uninterrupted operation of both satellites, and simultaneous tasking opportunities are limited. The observed equivalence in coverage should therefore be regarded as a demonstration of feasibility under ideal conditions, rather than as evidence of equal operational value.

5.3 Imaging Realism

The rendering pipeline demonstrates that off-nadir acquisitions degrade both geometric and radiometric quality, but several simplifications remain. While Mitsuba is physically based and captures geometric projection and radiometric effects such as BRDF and sun glint, it does not yet represent sensor- and platform-specific degradations. These include optical point-spread broadening with look angle, integration-time motion blur from finite exposures, and residual pointing jitter, all of which can further reduce effective resolution. Atmospheric variability is also not included, and cloud cover, a dominant maritime failure mode, is absent.

Wave modelling is simplified to sinusoidal patterns without coupling to wind fields or oceanographic variability, and the atmosphere is represented with a standard clear-sky profile that does not capture aerosol or humidity variation. Validation against real imagery is ongoing: spectral radiance values and glint patterns are currently being compared with Φsat-2 acquisitions. Once this validation is completed and additional scene and sensor effects are incorporated, the synthetic datasets will provide a more robust basis for AI training and evaluation.

5.4 Operational Considerations

The study assumes onboard detection and cueing but does not yet account for inter-satellite crosslinks, Cue-to-ground latency, or constraints from downlink bandwidth, power availability, or thermal duty cycles. These factors could significantly influence relative performance. Larger constellations benefit from redundancy and parallelism, which may mitigate bottlenecks in downlink or retargeting. Explicitly modelling these operational aspects would allow more comprehensive comparisons between minimal Tip and Cue architectures and larger fleets.

5.5 Benchmarking and Evaluation Metrics

The proposed benchmark links mission-level metrics (coverage, latency, off-nadir statistics) with AI-level metrics (mAP, precision, recall). To reflect operational usefulness more accurately, it must also capture costs and outcomes at the mission level. Two refinements are particularly important. First, false cue cost: the number of Cue retargetings wasted on false alarms, which will become critical once Tip detections are modelled realistically. Second, resurfacing-aware detections: for intermittent targets such as whales, detections should be tied to visibility cycles to avoid overestimating success rates. Event-level outcomes, such as confirming the presence of a pod or vessel track within a time horizon, are also needed to move from object counts to mission-relevant measures.

5.6 Considerations for Mission Design

Several considerations for mission design follow from these results. Off-nadir limits should not be set purely on geometric grounds, but explicitly linked to AI detection reliability, balancing latency against performance degradation. Dataset diversity emerges as a requirement: training must include realistic variations in angle, illumination, sea state, and radiometry, with emphasis on challenging cases such as strong sun glint and low-contrast conditions. Tasking strategies should also be reconsidered: immediate acquisition is not always optimal, and delaying observations to achieve a more favorable geometry may yield higher detection reliability. Finally, the architectural trade-off remains: Tip and Cue reduces satellite count and cost, but large constellations provide resilience through redundancy and parallel tasking. The right balance will depend on whether efficiency or robustness is prioritized.

5.7 Priorities for Future Work

Several priorities remain to strengthen the framework. First, realistic Tip detection models, based on actual Tipping satellite data, must be introduced. Second,

environmental effects such as whale resurfacing cycles, cloud cover, and sea-state variability need to be modelled to avoid optimistic detection counts. Third, imaging realism should be expanded to include MTF degradation, jitter, motion blur, and variable atmospheric conditions, while synthetic renderings are validated more extensively against real imagery beyond Φsat-2. Operational aspects such as crosslinks, downlink scheduling, and power budgets also require integration, as does the development of synthetic-real hybrid datasets, since current availability remains limited. Finally, case studies should be extended to more persistent maritime targets such as vessels, oil spills, icebergs, or marine litter, which are less sensitive to resurfacing assumptions and of direct operational relevance.

Integrating AI evaluation into the mission simulations will then enable direct comparison of detection throughput and constellation efficiency across architectures, fulfilling the intended end-to-end benchmark.

5.8 Outlook

The framework currently demonstrates feasibility rather than efficiency. A minimal Tip and Cue architecture shows strong potential to reduce satellite count compared to a large Walker constellation, but only a full end-to-end benchmark that incorporates orbital dynamics, imaging physics, operational constraints, and AI detection performance will provide definitive evidence of its advantages. Establishing such a benchmark is a realistic and necessary next step toward guiding the design of future AI-enabled Earth observation missions.

6. Conclusions

This work introduced the first end-to-end framework for AI-based Tip and Cue missions, combining orbital dynamics, tasking logic, and stabilized pointing with physics-based rendering that captures off-nadir and radiometric effects. By linking mission simulation with dataset generation, the framework enables evaluation of how orbital design and imaging conditions impact AI detection performance under realistic constraints.

The whale detection case study showed that a single Tip and Cue pair can match the coverage of a 16-satellite Walker-Delta constellation, achieving 54 vs. 52 detections in 24 hours. This efficiency, however, comes at the cost of longer tasking delays and degraded image quality from frequent off-nadir slews. The findings confirm that coverage alone is not a sufficient performance metric, latency, geometry, and image quality must also be considered.

The rendering pipeline demonstrated that physically based simulation can reproduce geometric and radiometric degradations relevant for onboard AI, modelling sun glints, but limitations remain, including simplified sea state, omission of clouds, and lack of sensor-specific artifacts. Ongoing validation with Φsat-2 data will improve realism and strengthen dataset reliability.

The main contribution of this work is establishing a benchmark that unifies mission-level and AI-level performance. While whale detection served as the demonstration, the framework generalizes to other maritime and environmental monitoring tasks such as vessel tracking, oil spill detection, and marine litter mapping. Future work will extend the framework to full end-to-end mission simulations, where onboard AI models are trained on synthetic off-nadir imagery and used directly for whale detection. This will allow performance comparisons to capture not only orbital dynamics and tasking but also the actual reliability of AI detections under actual imaging conditions.

Ultimately, this approach provides the basis for designing AI-enabled Earth observation missions that balance responsiveness, efficiency, and robustness, guiding the trade-offs between minimal Tip and Cue architectures and larger constellations.

Code Availability

The software is available open-source through GitHub - AI-Based Tip and Cue, with the first release planned for December 2025.

Acknowledgements

The author thanks the European Space Agency Φ-lab for providing research support, technical resources, and collaborative guidance during the development of this work.

References

[1] B. Huang, H. Zhang, H. Song, J. Wang, and C. Song, "Unified fusion of remote-sensing imagery: Generating simultaneously high-resolution synthetic spatial-temporal-spectral earth observations," Remote Sensing Letters, vol. 4, no. 6, pp. 561-569, 2013. doi: https://doi.org/10.1080/2150704X.2013.769283

[2] S. Vellas and M. Bernou, "IDEA I-2022-00382 - Dual Camera Satellite with On-Board AI-Based Decision-Making Capabilities," OHB-Hellas (Prime), FORTH (Subcontractor), Germany, Executive Summary Report 4000138071, Dec. 2022, Prepared for the ESA Cognitive Cloud Computing in Space campaign. [Online]. Available: https://nebula.esa.int/sites/default/files/neb_study/2675/C4000138110ExS.pdf.

[3] A. Alzubairi, A. Tameem, and B. Kada, "Spacecraft formation flying orbital control for Earth observation mission," Scientific African, vol. 26, e02391, 2024, issn: 2468-2276. doi: https://doi.org/10.1016/j.sciaf.2024.e02391.

[4] V. Růžička, G. Mateo-Garcia, A. Vaughan, L. Guanter, L. Gómez-Chova, and J. Parr, "STARCOP: Automated and self-improving follow-up verification of detrimental human-activity from LEO," European Space Agency (ESA), Technical Report Starcop 1-2022-00380, Nov. 2022, Published as part of the ESA Cognitive Cloud Computing in Space Campaign. Implemented as ESA Initial Support for Innovation. Agreement/PO number: 4000138110. [Online]. Available: https://nebula.esa.int/sites/default/files/neb_study/2675/C4000138110ExS.pdf.

[5] M. I. Ali. "Tip and cue technique for satellite monitoring of moving objects," Accessed: Dec. 31, 2024. [Online]. Available: https://www.iceye.com/blog/tip- and- cue-technique-for-efficient-near-real-time-satellite-monitoring-of-moving-objects.

[6] C. R. Post, "Towards automation of tipping and cueing between small satellites in a constellation," AFIT-ENG-MS-17-M-061. Approved for public release; distribution unlimited., Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, Mar. 2017

[7] P. Gómez, J. Östman, V. M. Shreenath, and G. Meoni, "PASEOS Simulates the Environment for Operating multiple Spacecraft," arXiv preprint arXiv:2302.02659 [cs.DC], 2023.

[8] W. Jakob, S. Speierer, N. Roussel, M. Nimier-David, D. Vicini, T. Zeltner, B. Nicolet, M. Crespo, V. Leroy, and Z. Zhang, *Mitsuba 3 renderer*, version 3.0.1, 2022. [Online]. Available: https://mitsuba-renderer.org

[9] C. A. Gueymard, *SMARTS: Simple Model of the Atmospheric Radiative Transfer of Sunshine*. National Renewable Energy Laboratory. [Online]. Available: https://www.nrel.gov/grid/solar-resource/smarts

[10] S. Huang, Z. Lu, X. Cun, Y. Yu, X. Zhou, and X. Shen, "DEIM: DETR with Improved Matching for Fast Convergence," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2025.

[11] Cubaynes, H.C., Fretwell, P.T. Whales from space dataset, an annotated satellite image dataset of whales for training machine learning models. *Sci Data* **9**, 245 (2022). https://doi.org/10.1038/s41597-022-01377-4

I

Appendix I - Planning

This appendix provides the project planning material that supported the execution of the thesis. It includes the work-package structure, workflow diagrams, and Gantt chart used to organize the development, validation, and experimentation phases of the project. The appendix is included for completeness and to document how the research was structured over time.

I.1. Work Packages

The thesis was structured into five work packages (WP), as summarized in Table I.1. The structure reflects the development path of the project: from tool exploration, to benchmark definition, to software integration, and finally to experimentation and evaluation.

The work started with WP1: Familiarization. During this phase, PASEOS was explored for orbit propagation and mission-level simulation, the off-nadir rendering pipeline was analysed and selected, and Radiative Transfer Modelling (RTM) tools were evaluated. The goal was, besides understanding these tools individually, to assess their compatibility for integration within a single simulation framework. Each component was tested separately before integration.

In WP2: Conceptualization, the performance benchmark was defined. This included the definition of mission-level, detection-level, and system-level metrics, as well as fixed reference conditions for orbit, sensor parameters, and simulation duration. Based on this benchmark, the demonstration experiments were specified. This phase translated conceptual research questions into concrete simulation requirements.

WP3: Development focused on integrating the selected tools into a functioning simulation chain. PASEOS was coupled to the off-nadir geometry computation and MITSUBA-based image rendering. Stabilization limits, pointing constraints, delays, and onboard AI behaviour were implemented in the orbital simulation framework. Evaluation scripts were developed in parallel to ensure that each detection model produced reproducible quantitative outputs. The integrated framework was validated using simplified scenarios before moving to full experiments.

In WP4: Experimentation, datasets were generated using the automated pipeline. Off-nadir distortions, environmental effects, and observation constraints were varied, after which onboard AI detection models were trained and evaluated on rendered Cue imagery. Simulation runs were executed for different orbital and temporal configurations, and results were fused and evaluated according to the benchmark metrics.

WP0: Project Management ran in parallel throughout the thesis. This included documentation, report writing, version control, milestone preparation, and adaptation of the planning with technical progress under circumstances.

I.1.1. Work Breakdown Structure

The Work Breakdown Structure is shown in Figure I.1. The Work Flow Diagram in Figure I.2 illustrates the dependency between the work packages. Those diagrams were used as a guiding structure rather than a

fixed constraint, as iterations in the development, integration, and evaluation phase required parallel work.

Table I.1: The AI-based Tip and Cue Master thesis work packages with their description, input, and output.

Category	Work Package	Description	Input	Output
WP 0 Project Management	WP 0.1	Write the report and white paper	TU Delft report writing requirements	Master thesis report, white paper
	WP 0.2	Update the planning	Graduation procedure guidelines	Updated Gantt chart
	WP 0.3	Prepare milestone presentations	Current progress	Review presentation slides and delivery
	WP 0.4	Manage progress meetings	Current progress	Meeting agenda, planning, and management
WP 1 Familiarization	WP 1.1	Familiarize with PASEOS	PASEOS GitHub	Reliable Python environment, PASEOS understanding
	WP 1.2	Familiarize with off-nadir imaging software	Off-nadir imaging tools	Off-nadir tool selection, testing, and understanding
	WP 1.3	Familiarize with Radiative Transfer Modelling software	RTM simulation tools	RTM tool selection, testing, and understanding
WP 2 Conceptualization	WP 2.1	Establish performance benchmark	AI model performance metrics, PASEOS understanding	Performance benchmark definition
	WP 2.2	Define demonstration experiments	Benchmark definition, RTM, off-nadir, PASEOS understanding	Experiment definition
WP 3 Development	WP 3.1	Integrate PASEOS and off-nadir imaging software	Off-nadir tool selection, PASEOS	Integrated PASEOS and off-nadir imaging software
	WP 3.2	Integrate PASEOS and Radiative Transfer Modelling software	RTM tool selection, PASEOS	Integrated PASEOS and RTM software
	WP 3.3	Establish performance evaluation scripts	Experiment definition	Experiment evaluation scripts
	WP 3.4	Validate the integrated software	Integrated PASEOS, off-nadir, and RTM software, Evaluation scripts	Validated software
WP 4 Experimentation	WP 4.1	Generate experiment datasets	Validated software	Dataset with disturbances
	WP 4.2	Develop experiment AI models	Dataset with disturbances, AI model availability	Trained AI model
	WP 4.3	Execute experiment simulations	Dataset with disturbances, trained AI model, Validated software	Experiment results
	WP 4.4	Evaluate experiment results	Experiment results	Result evaluation and conclusion

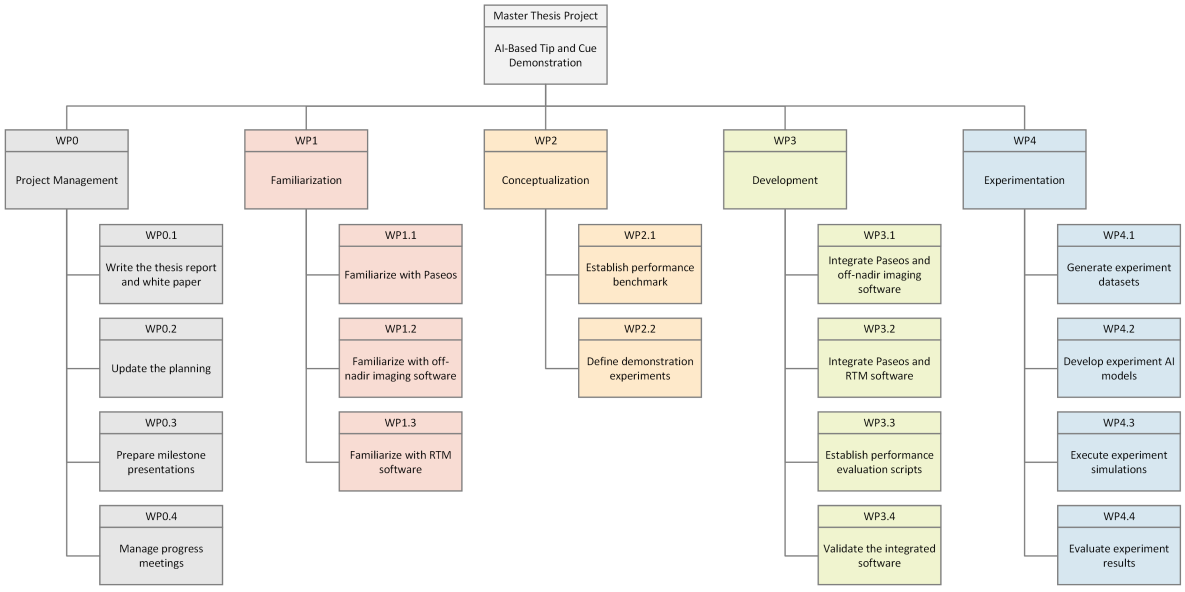


Figure I.1: The Work Breakdown Structure for the AI-based Tip and Cue master thesis demonstration project.

I.1.2. Work Flow Diagram

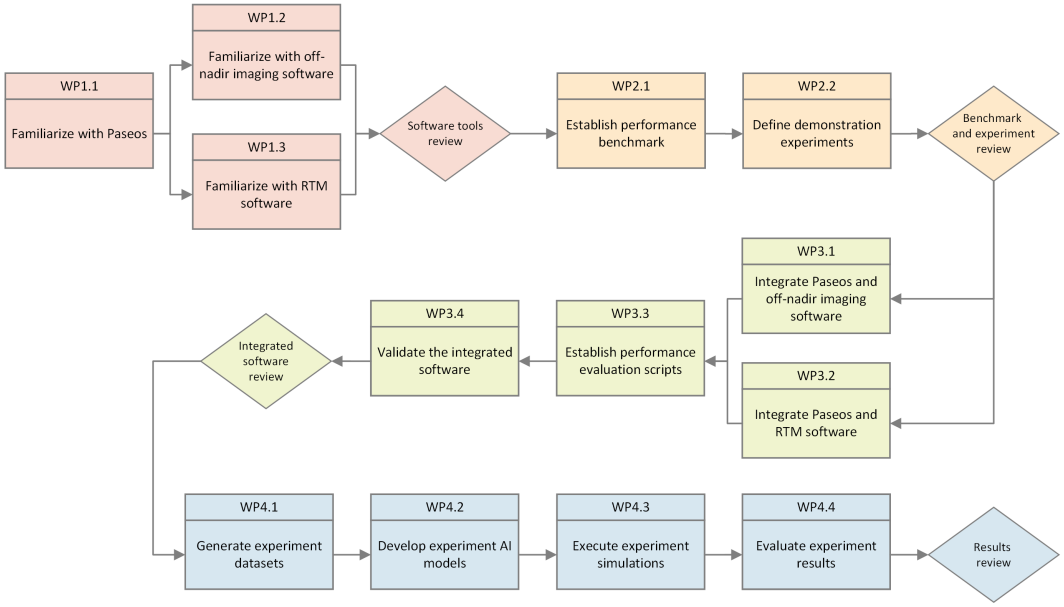


Figure I.2: The Work Flow Diagram for the AI-based Tip and Cue master thesis demonstration project highlighting the order of the work packages and milestones. The Project Management workpackage is not shown, as this process runs in parallel throughout the software development.

I.2. Gantt Chart

The Gantt chart was derived from the work packages in Table I.2, the targeted deadlines in Table I.3, and planned vacation periods in Table I.4.

Margins were included after concluding phases. Initially, these buffers were planned for integration challenges, remote computing set-up, evaluation iterations, or other troubleshooting and adjustments during the implementation.

The original planned duration was 30 weeks, starting in mid-April 2025 and ending with the defence in December 2025. The Gantt chart records the initial project planning and is included as a historical project-management record.

Table I.2: Project plan with work packages, durations, and start dates.

WP	Description	Duration (weeks)	Start date
1	**Familiarization**		
1.1	Familiarize with PASEOS	1	ma 14-04-25
1.2	Familiarize with off-nadir imaging software	1	ma 28-04-25
1.3	Familiarize with RTM modelling software	1	ma 05-05-25
	Margin	0.4	ma 12-05-25
2	**Conceptualization**		
2.1	Establish performance benchmark	1	ma 12-05-25
2.2	Define demonstration experiments	1	do 07-08-25
	Margin	0.6	do 14-08-25
3	**Development**		
3.1	Integrate PASEOS and off-nadir imaging software	2	ma 05-05-25
3.2	Integrate PASEOS and RTM software	2	ma 02-06-25
3.3	Establish performance evaluation scripts	1	ma 18-08-25
3.4	Validate the integrated software	1	ma 01-09-25
	Margin	1	ma 08-09-25
4	**Experimentation**		
4.1	Generate experiment datasets	2	ma 07-07-25
4.2	Develop experiment AI models	3	ma 27-10-25
4.3	Execute experiment simulations	2	ma 15-09-25
4.4	Evaluate experiment results	1	ma 10-11-25
	Margin	1	ma 17-11-25
5	**Improvements**		
5.1	Add more applications	1	ma 18-08-25
5.2	Improve AI models / simulation	1	ma 08-09-25
5.3	Perform more experiments	1	ma 17-11-25
	Margin	1	ma 24-11-25
0	**Project Management**		
0.1	Write the thesis report and white paper	2	ma 17-11-25
0.2	Update the planning	-	ma 24-11-25
0.3	Prepare milestone presentations	2	ma 25-08-25
0.4	Manage progress meetings	-	ma 22-12-25
	Total	**30**	

Table I.3: Key deadlines for the master thesis

Deadline	Week	Deadline
Midterm review	18	ma 25-08-25
Draft thesis hand-in	24	ma 10-11-25
Green light review	26	ma 24-11-25
Final thesis hand-in	28	ma 08-12-25
Thesis defence	30	ma 22-12-25

Table I.4: Planned vacations during the thesis period with approximate start date and duration.

Vacations	Duration (weeks)	Start date
Easter Italy train trip with Yuri	1	ma 21-04-25
Dolomites with ESA mountain club	1	do 02-07-26
International Astronautical Congress Sydney	2	do 25-09-25
World Biggest Analog	2	za 11-10-25

Master Thesis Planner

2025/2026

	M	D	W	T	F	S	S
APR		1	2	3	4	5	6
	7	8	9	10	11	12	13
	14	15	16	17	18	19	20
	21	22	23	24	25	26	27
	28	29	30	1	2	3	4
MAY	5	6	7	8	9	10	11
	12	13	14	15	16	17	18
	19	20	21	22	23	24	25
	26	27	28	29	30	31	1
JUN	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29
JUL	30	1	2	3	4	5	6
	7	8	9	10	11	12	13
	14	15	16	17	18	19	20
	21	22	23	24	25	26	27
	28	29	30	31	1	2	3
AUG	4	5	6	7	8	9	10
	11	12	13	14	15	16	17
	18	19	20	21	22	23	24
	25	26	27	28	29	30	31
SEP	1	2	3	4	5	6	7
	8	9	10	11	12	13	14
	15	16	17	18	19	20	21
	22	23	24	25	26	27	28
	29	30	1	2	3	4	5
OCT	6	7	8	9	10	11	12
	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29	30	31	1	2
NOV	3	4	5	6	7	8	9
	10	11	12	13	14	15	16
	17	18	19	20	21	22	23
	24	25	26	27	28	29	30
DEC	1	2	3	4	5	6	7
	8	9	10	11	12	13	14
	15	16	17	18	19	20	21
	22	23	24	25	26	27	28
JAN	29	30	31	1	2	3	4
	5	6	7	8	9	10	11
	12	13	14	15	16	17	18
	19	20	21	22	23	24	25
	26	27	28	29	30	31	1
FEB	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	1
MAR	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29
APR	30	31	1	2	3	4	5
	6	7	8	9	10	11	12
	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29	30	1	2	3
MAY	4	5	6	7	8	9	10
	11	12	13	14	15	16	17
	18	19	20	21	22	23	24

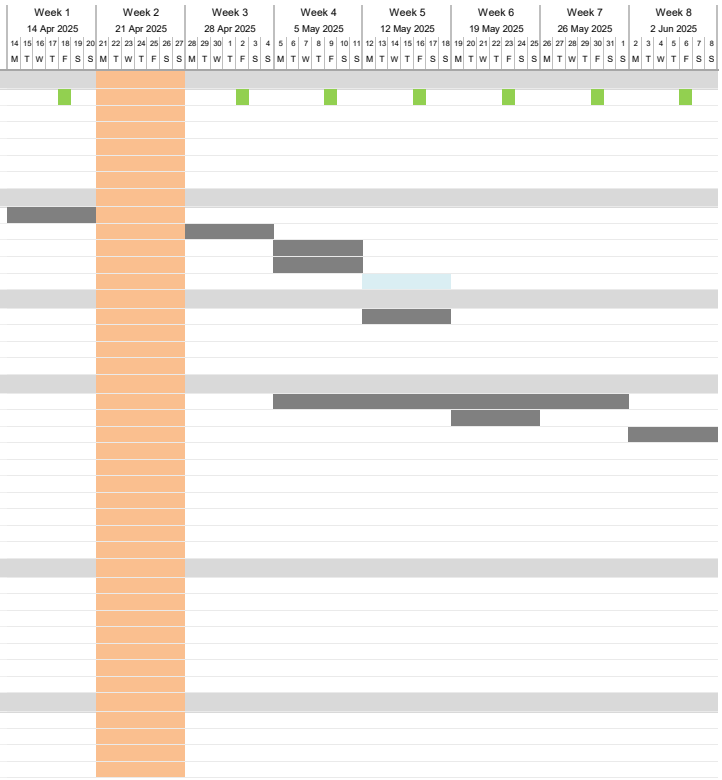
EVENT DATES	
May 17	Wedding Eline & Martijn
July 19	Wedding Tom & Josefien

TIME OFF	
April 1	Move to Frascati!
April 21 - 27	Easter Italy train trip w/ Yuri
May 16 - 18	Fly back home around wedding Eline (May 17)
July 2 - 9	Dolomites with ESA mountain club + Move back to NL
July 16	Space Cowboys podcast
August 27	Space Cowboys podcast
~~September 2 - 9~~	~~Albania with Anica~~
~~September 25 - 30~~	~~International Astronautical Congress Sydney~~
October 1 - 4	International Astronautical Congress Sydney
~~October 6 - 10~~	~~International Astronautical Congress Sydney~~
October 11 - 26	World Biggest Analog

MILESTONES	
April 14	TH WK 1 - Start Master Thesis
August 25	TH WK 18 - Mid Term Review
November 10	TH WK 24 - Hand-in Draft thesis (T -14 days)
November 24	TH WK 26 - Green Light Review
December 1	TH WK 27 - Request examination (T -20 days)
December 8	TH WK 28 - Hand-in Final Thesis (T -14 days)
December 22	TH WK 30 - Thesis defence

Project Start Date: 14-04-2025 (Monday)
Project Lead:
Display Week: 1

WP	TASK	START	END	DAYS	% DONE	WORK DAYS
0	**Project Management**					
0.1	Write the thesis report and white paper	Mon 17-11-25	Sun 30-11-25	14	100%	10
0.2	Update the planning	-	-	-	-	-
0.3	Prepare midterm review	Mon 18-08-25	Sun 24-08-25	7	100%	
0.4	Write thesis report	Mon 24-11-25	Sun 07-12-25	14	100%	
0.5	Prepare final presentation	Mon 08-12-25	Sun 21-12-25	14	100%	10
0.6	Manage progress meetings	-	-	-	-	-
1	**Familiarization**					
1.1	Familiarize with Paseos	Mon 14-04-25	Sun 20-04-25	7	100%	5
1.2	Explore software options for off-nadir imaging and radiance modelling	Mon 28-04-25	Sun 04-05-25	7	100%	5
1.3	Familiarize with Mitsuba (Physically-based rendering)	Mon 05-05-25	Sun 11-05-25	7	100%	5
1.4	Familiarize with SMARTS (RTM modelling)	Mon 05-05-25	Sun 11-05-25	7	100%	5
	Margin	Mon 12-05-25	Sun 18-05-25	7	-	5
2	**Conceptualization**					
2.1	Start thinking of benchmarks	Mon 12-05-25	Sun 18-05-25	7	100%	5
2.2	Finalize performance benchmarks	Thu 07-08-25	Sun 10-08-25	4	100%	2
2.3	Define demonstration experiments	Thu 07-08-25	Sun 10-08-25	4	100%	2
	Margin		-	7	-	-
3	**Development**					
3.1	Implement off-nadir imaging in Mitsuba	Mon 05-05-25	Sun 01-06-25	28	100%	20
3.2	Generate artificial DEM	Mon 19-05-25	Sun 25-05-25	7	100%	5
3.3	Implement radiance modelling in Mitsuba	Mon 02-06-25	Sun 22-06-25	21	100%	15
3.4	Implement RTM model with SMARTS and integrate	Mon 23-06-25	Sun 06-07-25	14	100%	10
3.5	Integrate Paseos and off-nadir software	Mon 07-07-25	Sun 13-07-25	7	100%	5
3.6	Add satellite stabilization part	Mon 04-08-25	Sun 17-08-25	14	100%	10
3.7	Integrate all software	Mon 28-07-25	Sun 03-08-25	7	100%	5
3.8	Establish performance evaluation scripts	Mon 18-08-25	Sun 24-08-25	7	100%	5
3.9	Verify the integrated software	Mon 01-09-25	Fri 05-09-25	5	100%	5
	Margin		-	7	-	-
4	**Experimentation**					
4.1	Format dataset to COCO	Mon 07-07-25	Sun 13-07-25	7	100%	5
4.2	Set up remote computing and test DEIM model training	Mon 14-07-25	Sun 27-07-25	14	100%	10
4.3	Obtain off-nadir distribution + prepare datasets	Mon 01-09-25	Sun 14-09-25	14	100%	10
4.4	Train models and perform model experiments	Mon 27-10-25	Sun 16-11-25	21	100%	15
4.5	Execute experiment simulations	Mon 15-09-25	Wed 24-09-25	10	100%	8
4.6	Evaluate experiment results	Mon 10-11-25	Sun 16-11-25	7	100%	5
	Margin		-	7	-	-
5	**Validation**					
5.1	Validate Mitsuba radiation model	Mon 18-08-25	Sun 31-08-25	14	100%	10
5.2	Validate mission simulation software	Mon 08-09-25	Sun 14-09-25	7	100%	5
5.3	Perform more experiments	Mon 17-11-25	Sun 23-11-25	7	100%	5
	Margin		-		100%	-

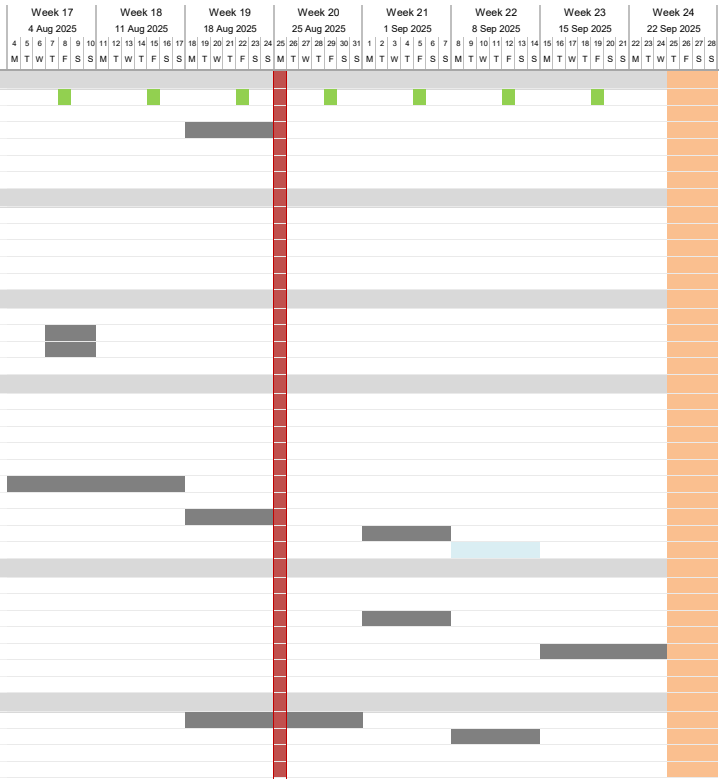


Project Start Date: 14-04-2025 (Monday)
Project Lead:
Display Week: 9

WP	TASK	START	END	DAYS	% DONE	WORK DAYS
0	**Project Management**					
0.1	Write the thesis report and white paper	Mon 17-11-25	Sun 30-11-25	14	100%	10
0.2	Update the planning	-	-	-	-	-
0.3	Prepare midterm review	Mon 18-08-25	Sun 24-08-25	7	100%	
0.4	Write thesis report	Mon 24-11-25	Sun 07-12-25	14	100%	
0.5	Prepare final presentation	Mon 08-12-25	Sun 21-12-25	14	100%	10
0.6	Manage progress meetings	-	-	-	-	-
1	**Familiarization**					
1.1	Familiarize with Paseos	Mon 14-04-25	Sun 20-04-25	7	100%	5
1.2	Explore software options for off-nadir imaging and radiance modelling	Mon 28-04-25	Sun 04-05-25	7	100%	5
1.3	Familiarize with Mitsuba (Physically-based rendering)	Mon 05-05-25	Sun 11-05-25	7	100%	5
1.4	Familiarize with SMARTS (RTM modelling)	Mon 05-05-25	Sun 11-05-25	7	100%	5
	Margin	Mon 12-05-25	Sun 18-05-25	7	-	5
2	**Conceptualization**					
2.1	Start thinking of benchmarks	Mon 12-05-25	Sun 18-05-25	7	100%	5
2.2	Finalize performance benchmarks	Thu 07-08-25	Sun 10-08-25	4	100%	2
2.3	Define demonstration experiments	Thu 07-08-25	Sun 10-08-25	4	100%	2
	Margin		-	7	-	-
3	**Development**					
3.1	Implement off-nadir imaging in Mitsuba	Mon 05-05-25	Sun 01-06-25	28	100%	20
3.2	Generate artificial DEM	Mon 19-05-25	Sun 25-05-25	7	100%	5
3.3	Implement radiance modelling in Mitsuba	Mon 02-06-25	Sun 22-06-25	21	100%	15
3.4	Implement RTM model with SMARTS and integrate	Mon 23-06-25	Sun 06-07-25	14	100%	10
3.5	Integrate Paseos and off-nadir software	Mon 07-07-25	Sun 13-07-25	7	100%	5
3.6	Add satellite stabilization part	Mon 04-08-25	Sun 17-08-25	14	100%	10
3.7	Integrate all software	Mon 28-07-25	Sun 03-08-25	7	100%	5
3.8	Establish performance evaluation scripts	Mon 18-08-25	Sun 24-08-25	7	100%	5
3.9	Verify the integrated software	Mon 01-09-25	Fri 05-09-25	5	100%	5
	Margin		-	7	-	-
4	**Experimentation**					
4.1	Format dataset to COCO	Mon 07-07-25	Sun 13-07-25	7	100%	5
4.2	Set up remote computing and test DEIM model training	Mon 14-07-25	Sun 27-07-25	14	100%	10
4.3	Obtain off-nadir distribution + prepare datasets	Mon 01-09-25	Sun 14-09-25	14	100%	10
4.4	Train models and perform model experiments	Mon 27-10-25	Sun 16-11-25	21	100%	15
4.5	Execute experiment simulations	Mon 15-09-25	Wed 24-09-25	10	100%	8
4.6	Evaluate experiment results	Mon 10-11-25	Sun 16-11-25	7	100%	5
	Margin		-	7	-	-
5	**Validation**					
5.1	Validate Mitsuba radiation model	Mon 18-08-25	Sun 31-08-25	14	100%	10
5.2	Validate mission simulation software	Mon 08-09-25	Sun 14-09-25	7	100%	5
5.3	Perform more experiments	Mon 17-11-25	Sun 23-11-25	7	100%	5
	Margin		-		100%	-

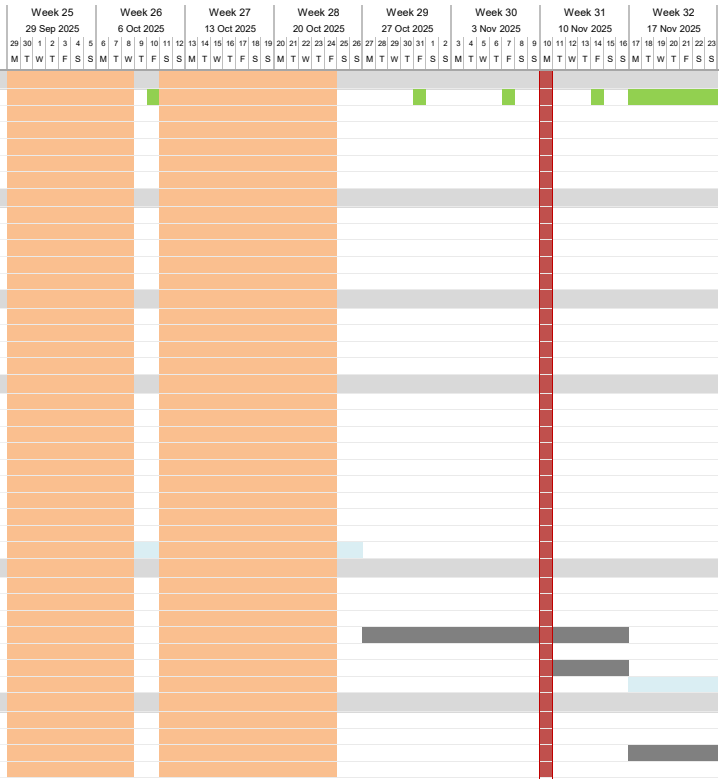
Project Start Date: 14-04-2025 (Monday)
Project Lead:
Display Week: 17

WP	TASK	START	END	DAYS	% DONE	WORK DAYS
0	**Project Management**					
0.1	Write the thesis report and white paper	Mon 17-11-25	Sun 30-11-25	14	100%	10
0.2	Update the planning	-	-	-	-	-
0.3	Prepare midterm review	Mon 18-08-25	Sun 24-08-25	7	100%	
0.4	Write thesis report	Mon 24-11-25	Sun 07-12-25	14	100%	
0.5	Prepare final presentation	Mon 08-12-25	Sun 21-12-25	14	100%	10
0.6	Manage progress meetings	-	-	-	-	-
1	**Familiarization**					
1.1	Familiarize with Paseos	Mon 14-04-25	Sun 20-04-25	7	100%	5
1.2	Explore software options for off-nadir imaging and radiance modelling	Mon 28-04-25	Sun 04-05-25	7	100%	5
1.3	Familiarize with Mitsuba (Physically-based rendering)	Mon 05-05-25	Sun 11-05-25	7	100%	5
1.4	Familiarize with SMARTS (RTM modelling)	Mon 05-05-25	Sun 11-05-25	7	100%	5
	Margin	Mon 12-05-25	Sun 18-05-25	7	-	5
2	**Conceptualization**					
2.1	Start thinking of benchmarks	Mon 12-05-25	Sun 18-05-25	7	100%	5
2.2	Finalize performance benchmarks	Thu 07-08-25	Sun 10-08-25	4	100%	2
2.3	Define demonstration experiments	Thu 07-08-25	Sun 10-08-25	4	100%	2
	Margin		-	7	-	-
3	**Development**					
3.1	Implement off-nadir imaging in Mitsuba	Mon 05-05-25	Sun 01-06-25	28	100%	20
3.2	Generate artificial DEM	Mon 19-05-25	Sun 25-05-25	7	100%	5
3.3	Implement radiance modelling in Mitsuba	Mon 02-06-25	Sun 22-06-25	21	100%	15
3.4	Implement RTM model with SMARTS and integrate	Mon 23-06-25	Sun 06-07-25	14	100%	10
3.5	Integrate Paseos and off-nadir software	Mon 07-07-25	Sun 13-07-25	7	100%	5
3.6	Add satellite stabilization part	Mon 04-08-25	Sun 17-08-25	14	100%	10
3.7	Integrate all software	Mon 28-07-25	Sun 03-08-25	7	100%	5
3.8	Establish performance evaluation scripts	Mon 18-08-25	Sun 24-08-25	7	100%	5
3.9	Verify the integrated software	Mon 01-09-25	Sun 07-09-25	7	100%	5
	Margin		-	7	-	-
4	**Experimentation**					
4.1	Format dataset to COCO	Mon 07-07-25	Sun 13-07-25	7	100%	5
4.2	Set up remote computing and test DEIM model training	Mon 14-07-25	Sun 27-07-25	14	100%	10
4.3	Obtain off-nadir distribution + prepare datasets	Mon 01-09-25	Sun 07-09-25	7	100%	5
4.4	Train models and perform model experiments	Mon 27-10-25	Sun 16-11-25	21	100%	15
4.5	Execute experiment simulations	Mon 15-09-25	Wed 24-09-25	10	100%	8
4.6	Evaluate experiment results	Mon 10-11-25	Sun 16-11-25	7	100%	5
	Margin		-	7	-	-
5	**Validation**					
5.1	Validate Mitsuba radiation model	Mon 18-08-25	Sun 31-08-25	14	100%	10
5.2	Validate mission simulation software	Mon 08-09-25	Sun 14-09-25	7	100%	5
5.3	Perform more experiments	Mon 17-11-25	Sun 23-11-25	7	100%	5
	Margin		-		100%	-



Project Start Date: 14-04-2025 (Monday)
Project Lead:
Display Week: 25

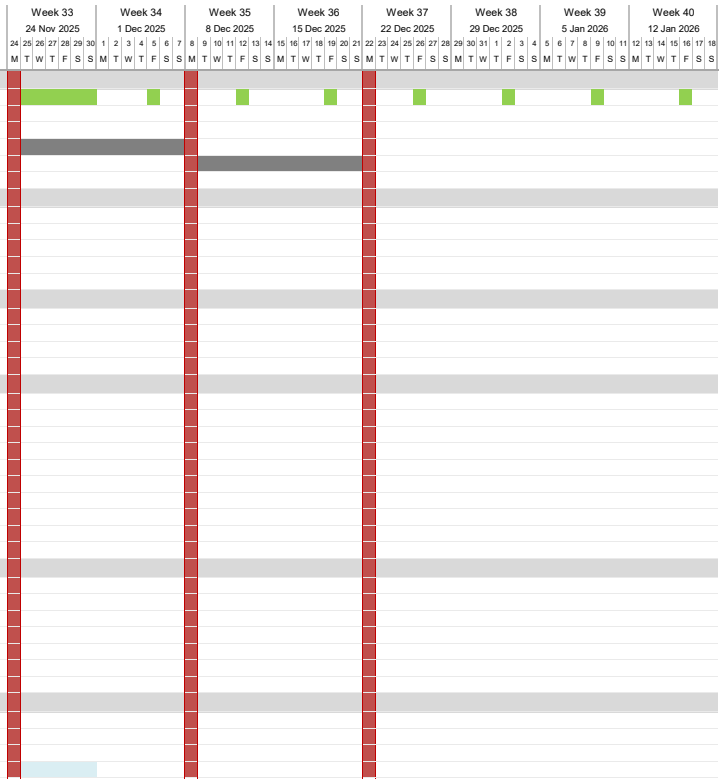
WP	TASK	START	END	DAYS	% DONE	WORK DAYS
0	**Project Management**					
0.1	Write the thesis report and white paper	Mon 17-11-25	Sun 30-11-25	14	100%	10
0.2	Update the planning	-	-	-	-	-
0.3	Prepare midterm review	Mon 18-08-25	Sun 24-08-25	7	100%	
0.4	Write thesis report	Mon 24-11-25	Sun 07-12-25	14	100%	
0.5	Prepare final presentation	Mon 08-12-25	Sun 21-12-25	14	100%	10
0.6	Manage progress meetings	-	-	-	-	-
1	**Familiarization**					
1.1	Familiarize with Paseos	Mon 14-04-25	Sun 20-04-25	7	100%	5
1.2	Explore software options for off-nadir imaging and radiance modelling	Mon 28-04-25	Sun 04-05-25	7	100%	5
1.3	Familiarize with Mitsuba (Physically-based rendering)	Mon 05-05-25	Sun 11-05-25	7	100%	5
1.4	Familiarize with SMARTS (RTM modelling)	Mon 05-05-25	Sun 11-05-25	7	100%	5
	Margin	Mon 12-05-25	Sun 18-05-25	7	-	5
2	**Conceptualization**					
2.1	Start thinking of benchmarks	Mon 12-05-25	Sun 18-05-25	7	100%	5
2.2	Finalize performance benchmarks	Thu 07-08-25	Sun 10-08-25	4	100%	2
2.3	Define demonstration experiments	Thu 07-08-25	Sun 10-08-25	4	100%	2
	Margin		-	7	-	-
3	**Development**					
3.1	Implement off-nadir imaging in Mitsuba	Mon 05-05-25	Sun 01-06-25	28	100%	20
3.2	Generate artificial DEM	Mon 19-05-25	Sun 25-05-25	7	100%	5
3.3	Implement radiance modelling in Mitsuba	Mon 02-06-25	Sun 22-06-25	21	100%	15
3.4	Implement RTM model with SMARTS and integrate	Mon 23-06-25	Sun 06-07-25	14	100%	10
3.5	Integrate Paseos and off-nadir software	Mon 07-07-25	Sun 13-07-25	7	100%	5
3.6	Add satellite stabilization part	Mon 04-08-25	Sun 17-08-25	14	100%	10
3.7	Integrate all software	Mon 28-07-25	Sun 03-08-25	7	100%	5
3.8	Establish performance evaluation scripts	Mon 18-08-25	Sun 24-08-25	7	100%	5
3.9	Verify the integrated software	Mon 01-09-25	Sun 07-09-25	7	100%	5
	Margin		-	7	-	-
4	**Experimentation**					
4.1	Format dataset to COCO	Mon 07-07-25	Sun 13-07-25	7	100%	5
4.2	Set up remote computing and test DEIM model training	Mon 14-07-25	Sun 27-07-25	14	100%	10
4.3	Obtain off-nadir distribution + prepare datasets	Mon 01-09-25	Sun 07-09-25	7	100%	5
4.4	Train models and perform model experiments	Mon 27-10-25	Sun 16-11-25	21	100%	15
4.5	Execute experiment simulations	Mon 15-09-25	Wed 24-09-25	10	100%	8
4.6	Evaluate experiment results	Mon 10-11-25	Sun 16-11-25	7	100%	5
	Margin		-	7	-	-
5	**Validation**					
5.1	Validate Mitsuba radiation model	Mon 18-08-25	Sun 31-08-25	14	100%	10
5.2	Validate mission simulation software	Mon 08-09-25	Sun 14-09-25	7	100%	5
5.3	Perform more experiments	Mon 17-11-25	Sun 23-11-25	7	100%	5
	Margin		-		100%	-



Project Start Date: 14-04-2025 (Monday) | Display Week: 33

Project Lead:

WP	TASK	START	END	DAYS	% DONE	WORK DAYS
0	**Project Management**					
0.1	Write the thesis report and white paper	Mon 17-11-25	Sun 30-11-25	14	100%	10
0.2	Update the planning	-	-	-	-	-
0.3	Prepare midterm review	Mon 18-08-25	Sun 24-08-25	7	100%	
0.4	Write thesis report	Mon 24-11-25	Sun 07-12-25	14	100%	
0.5	Prepare final presentation	Mon 08-12-25	Sun 21-12-25	14	100%	10
0.6	Manage progress meetings	-	-	-	-	-
1	**Familiarization**					
1.1	Familiarize with Paseos	Mon 14-04-25	Sun 20-04-25	7	100%	5
1.2	Explore software options for off-nadir imaging and radiance modelling	Mon 28-04-25	Sun 04-05-25	7	100%	5
1.3	Familiarize with Mitsuba (Physically-based rendering)	Mon 05-05-25	Sun 11-05-25	7	100%	5
1.4	Familiarize with SMARTS (RTM modelling)	Mon 05-05-25	Sun 11-05-25	7	100%	5
	Margin	Mon 12-05-25	Sun 18-05-25	7	-	5
2	**Conceptualization**					
2.1	Start thinking of benchmarks	Mon 12-05-25	Sun 18-05-25	7	100%	5
2.2	Finalize performance benchmarks	Thu 07-08-25	Sun 10-08-25	4	100%	2
2.3	Define demonstration experiments	Thu 07-08-25	Sun 10-08-25	4	100%	2
	Margin		-	7	-	-
3	**Development**					
3.1	Implement off-nadir imaging in Mitsuba	Mon 05-05-25	Sun 01-06-25	28	100%	20
3.2	Generate artificial DEM	Mon 19-05-25	Sun 25-05-25	7	100%	5
3.3	Implement radiance modelling in Mitsuba	Mon 02-06-25	Sun 22-06-25	21	100%	15
3.4	Implement RTM model with SMARTS and integrate	Mon 23-06-25	Sun 06-07-25	14	100%	10
3.5	Integrate Paseos and off-nadir software	Mon 07-07-25	Sun 13-07-25	7	100%	5
3.6	Add satellite stabilization part	Mon 04-08-25	Sun 17-08-25	14	100%	10
3.7	Integrate all software	Mon 28-07-25	Sun 03-08-25	7	100%	5
3.8	Establish performance evaluation scripts	Mon 18-08-25	Sun 24-08-25	7	100%	5
3.9	Verify the integrated software	Mon 01-09-25	Sun 07-09-25	7	100%	5
	Margin		-	7	-	-
4	**Experimentation**					
4.1	Format dataset to COCO	Mon 07-07-25	Sun 13-07-25	7	100%	5
4.2	Set up remote computing and test DEIM model training	Mon 14-07-25	Sun 27-07-25	14	100%	10
4.3	Obtain off-nadir distribution + prepare datasets	Mon 01-09-25	Sun 07-09-25	7	100%	5
4.4	Train models and perform model experiments	Mon 27-10-25	Sun 16-11-25	21	100%	15
4.5	Execute experiment simulations	Mon 15-09-25	Wed 24-09-25	10	100%	8
4.6	Evaluate experiment results	Mon 10-11-25	Sun 16-11-25	7	100%	5
	Margin		-	7	-	-
5	**Validation**					
5.1	Validate Mitsuba radiation model	Mon 18-08-25	Sun 31-08-25	14	100%	10
5.2	Validate mission simulation software	Mon 08-09-25	Sun 14-09-25	7	100%	5
5.3	Perform more experiments	Mon 17-11-25	Sun 23-11-25	7	100%	5
	Margin		-		100%	-



For completeness, the actual execution of the project is shown in Figure I.3.

The execution clearly does not follow the original planning. The planning became more of a guidance than something to strictly follow, while still keeping the work breakdown as the intended structure.

Work happened when it could. In the evenings. In hospital waiting rooms. In between conversations. Not because that was ideal, but because it allowed me to be present where I needed to be, while still continuing the project to the capacity that was available.

Originally, the goal was that my mom could still see my graduation, which unfortunately changed. A new milestone later set a clear end point, and that became the reason to make the final push.

This project taught me more than technical execution. It taught flexibility and adaptation. It required setting priorities. And it required resilience, accepting that some weeks give less, and some require everything.

In space, no mission flies exactly as planned.

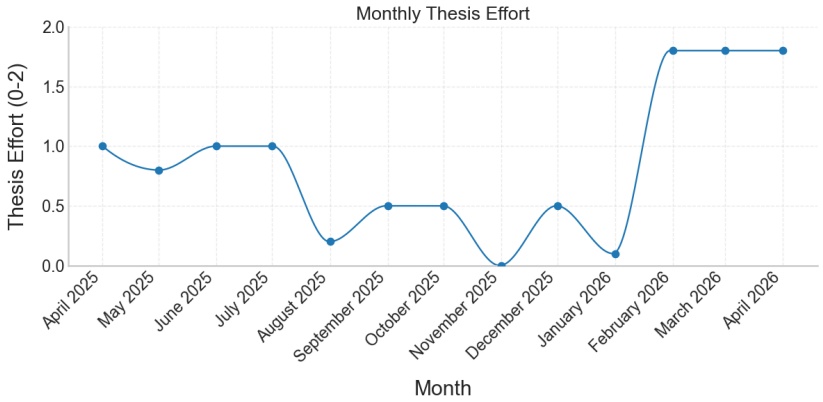


Figure I.3: And that is how this thesis was eventually completed.