Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional
Unidad Guadalajara

Clasificación a Nivel de Píxel de Imágenes Multi e Hiperespectrales Capturadas desde Satélites: Un Enfoque de Aprendizaje Profundo

Tesis que presenta:
Efraín Alberto Padilla Zepeda

para obtener el grado de:
Doctor en Ciencias

en la especialidad de:
Ingeniería Eléctrica

Directores de Tesis:
Dr. Deni Librado Torres Román
Dra. Raquel De Los Reyes López

Guadalajara, Jalisco
Julio de 2025

Clasificación a Nivel de Píxel de Imágenes Multi e Hiperespectrales Capturadas desde Satélites: Un Enfoque de Aprendizaje Profundo

Tesis de Doctorado en Ciencias
Ingeniería Eléctrica

Por:
Efraín Alberto Padilla Zepeda
Maestro en Ciencias en Ingeniería Eléctrica
Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional 2018-2020

Becario de SECIHTI, expediente no. 789304

Directores de Tesis:
Dr. Deni Librado Torres Román
Dra. Raquel De Los Reyes López

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional
Unidad Guadalajara, Julio de 2025.

Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional
Unidad Guadalajara

Pixel-Level Classification of Spaceborne Multi- and Hyperspectral Imagery: A Deep Learning Approach

A thesis presented by:
Efraín Alberto Padilla Zepeda

to obtain the degree of:
Doctor in Science
in the subject of:
Electrical Engineering

Thesis Advisors:
Dr. Deni Librado Torres Román
Dr. Raquel De Los Reyes López

Guadalajara, Jalisco July 2025

Pixel-Level Classification of Spaceborne Multi- and Hyperspectral Imagery: A Deep Learning Approach

Doctor of Science thesis
In Electrical Engineering

By:
Efraín Alberto Padilla Zepeda
Master of Science in Electrical Engineering
Centro de Investigación y de Estudios Avanzados del
Instituto Politécnico Nacional 2018-2020

Scholarship granted by SECIHTI, No. 789304

Thesis Advisors:
Dr. Deni Librado Torres Román
Dr. Raquel De Los Reyes López

Center for Research and Advanced Studies of the National Polytechnical Institute,
Guadalajara Unit, July 2025.

Author:

Efraín Alberto Padilla Zepeda
Cinvestav Guadalajara

Thesis Advisors:

Dr. Deni Librado Torres Román
Cinvestav Guadalajara

Dr. Raquel De Los Reyes López
German Aerospace Center (DLR)

In Collaboration with:

Dr. Kevin Alonso González
European Space Agency (ESA)

Resumen

Esta investigación tiene como objetivo mejorar los algoritmos de clasificación utilizados en la corrección atmosférica (AC), aprovechando las capacidades de extracción de características espectro-espaciales de los modelos de aprendizaje profundo (DL) de última generación para la clasificación a nivel de píxel en imágenes espectrales. El objetivo principal es avanzar en el estado del arte actual en preparación para la próxima generación de sensores hiperespectrales. La fase inicial de este trabajo se centra en la integración de algoritmos de clasificación basados en umbrales, ampliamente utilizados en teledetección multispectral, con redes neuronales convolucionales (CNN). Esta integración demuestra mejoras significativas en los resultados de clasificación al combinar enfoques tradicionales con técnicas modernas de aprendizaje profundo.

Como parte de esta investigación, se establecen lineamientos para criterios de etiquetado armonizados basados en parámetros físicos, esenciales para la creación de conjuntos de datos robustos de entrenamiento y prueba que integren múltiples fuentes de información, como etiquetas manuales y productos del sensor TROPOMI a bordo de la misión Sentinel-5P. Este trabajo utiliza el software PACO desarrollado por el DLR y realiza experimentos con datos de la misión multiespectral Sentinel-2 y de la misión hiperespectral EnMAP.

En el caso multiespectral, los modelos propuestos superan de forma consistente la línea base de PACO en una variedad de conjuntos de prueba, con mejoras de hasta 18.3 puntos porcentuales en el coeficiente de correlación de Matthews normalizado (nMCC). Las mayores mejoras se observaron en escenarios de clasificación complicados, especialmente en las clases de *sombras* y *suelo*.

En el caso hiperespectral, los resultados de clasificación fueron transformados a parámetros físicos mediante la comparación de las fracciones de nubosidad inferidas con las obtenidas por TROPOMI. Si bien las métricas globales de regresión mostraron un desempeño comparable entre los métodos basados en CNN y la referencia base de PACO, la principal contribución radica en el análisis de las fortalezas y debilidades de la metodología de validación propuesta. En particular, el uso de gráficos Precisión–Exactitud–Incertidumbre (APU) permitió una evaluación detallada del comportamiento del modelo en distintos rangos de fracciones de nubosidad, destacando las limitaciones en los datos de referencia y los efectos de los desajustes espaciales y temporales.

Al abordar los desafíos en la preparación de conjuntos de datos y utilizar modelos neuronales avanzados, este trabajo ofrece aportes valiosos para la obtención de productos de clasificación mejorados en misiones de teledetección multi- e hiper-espectral con una perspectiva de implementación global.

Abstract

This research aims to enhance the classification algorithms used in Atmospheric Correction (AC) by leveraging the spatial-spectral feature extraction capabilities of state-of-the-art Deep Learning (DL) models for pixel-level classification of spectral imagery. The primary objective is to advance the current state-of-the-art in preparation for the next generation of hyperspectral sensors. The initial phase of this work focuses on the integration of threshold-based masking algorithms, widely applied in multispectral remote sensing, with Convolutional Neural Networks (CNNs). This integration demonstrates significant improvements in classification outcomes, by combining traditional approaches with modern deep learning techniques.

As part of this research, guidelines are established for harmonized labeling criteria based on physical parameters, which are essential for creating robust training and testing datasets that incorporate multiple sources of information like manual labeling and products from the Tropospheric Monitoring Instrument (TROPOMI) sensor onboard the Sentinel-5P mission. This work uses the Python-based Atmospheric Correction (PACO) software developed by the German Aerospace Center “Deutsches Zentrum für Luft- und Raumfahrt” (DLR) and conducts experiments using data from the multispectral Sentinel-2 mission and the Environmental Mapping and Analysis Program (EnMAP), which captures hyperspectral imagery.

For the multispectral case, the proposed models consistently outperform the PACO baseline across a variety of testing datasets, showing improvements of up to 18.3 percentage points in normalized Matthew’s Correlation Coefficient (nMCC). The largest gains were observed in challenging classification scenarios, particularly in the *shadow* and *clear* classes.

In the hyperspectral case, classification results were translated into physical parameters by comparing the predicted cloud fractions with those from TROPOMI. While global regression metrics showed comparable performance between the CNN-based methods and the PACO baseline, the main contribution lies in the analysis of the strengths and weaknesses of the proposed validation methodology. In particular, the use of Accuracy, Precision and Uncertainty (APU) plots enabled a deep assessment of model behavior across different cloud fraction ranges, highlighting limitations in the reference data and the effects of spatial and temporal mismatches.

By addressing the challenges in dataset preparation and using cutting-edge neural network models, this work provides insights for obtaining improved classification products for multi- and hyper-spectral remote sensing missions with a global implementation perspective.

Acknowledgments

When I first thought about doing a Ph.D. in the middle of 2020, during a global pandemic, I never imagined I would be writing these acknowledgments while flying over the Atlantic Ocean in the way back to Guadalajara, Mexico, after spending time abroad and meeting people from around the world. I've been lucky to share this journey with many people. Although I can't mention everyone, I'm truly grateful to each of you.

This thesis was developed in two main phases: the first at Cinvestav Guadalajara, Mexico (January 2021 – July 2023), and the second in DLR Oberpfaffenhofen, Germany (August 2023 – June 2025), as part of a bi-national Ph.D. program.

First, I want to thank the people who made this possible—my supervisors:

Dr. Deni Torres, who also guided me during my master's studies. His advice and constant support helped me grow as a researcher. Thank you for always pushing me to understand things deeply and for your help with every part of the process, including all the paperwork.

Dr. Raquel de los Reyes, whose commitment to science is truly inspiring. Her energy, knowledge, and daily support made a big difference. I really appreciate the time she took to check in, solve doubts, and help me move forward. I hope to give back someday as much as she gave me.

Dr. Kevin Alonso, who was my first contact at EOC-DLR. I'm very thankful he believed in me from the beginning and supported a collaboration between our groups. His ideas and guidance helped shape this work.

I also want to thank Dr. Gustavo Martín del Campo and Dr. Joel Amao, who sparked my interest in this research area. Even though the original plan for a short stay at DLR didn't happen due to the pandemic, their support and connections were key to continuing my work.

Thanks to my colleagues at Cinvestav and DLR for always sharing their knowledge and advice. I also appreciated the simple everyday moments that made working together enjoyable.

To my friends: Sergio Serafín, for sharing his apartment when I first arrived in Munich and helping me find my own place. Mario Fuentes, for all his advice and support at DLR. Daniel Hernández, my friend since we started the master's. And Kevin Kühl, who helped me feel at home far from home. Thanks also to my sports friends for the bike rides, hikes, and football matches.

To my family—thank you for always being there. To my mom, Ruth Zepeda, whose blessing made me feel invincible. To my dad, Martín Padilla López, for teaching me to work hard and honestly. To my brother, Martín Padilla, my lifelong role model.

To my girlfriend, Montserrat Esparza-Díaz, thank you for your support from almost the very beginning. You shared the good moments and helped me through the hard ones. I'm especially grateful for your strength during the long months apart.

Thanks also to the organizations that supported this thesis:

I would like to thank the Secretariat of Science, Humanities, Technology and Innovation "Secretaría de Ciencia, Humanidades, Tecnología e Innovación" (Secihti), for the Ph.D. grant number 789304 and the Center for Research and Advanced Studies of the National Polytechnic Institute (Mexico) "Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional" (Cinvestav) for serving as the home institution for my doctoral studies.

I would also extend our gratitude to the German Academic Exchange Service "Deutscher Akademischer Austauschdienst" (DAAD) for supporting the research stay in Germany, funded by the Federal Ministry of Education and Research (Germany) "Bundesministerium für Bildung und Forschung" (BMBF) through DAAD fellowship number 57645446.

I thank the DLR for providing a workplace and supporting this research. Our appreciation extends to the Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities "Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften" (LRZ) for providing computational resources through Terrabyte.

Finally, thank you to Germany, for all the lessons I learned while living and working there, and to Mexico, for reminding me how proud I am to be Mexican.

Contents

Acronyms

AC Atmospheric Correction

ACIX Atmospheric Correction Inter-comparison eXercise

ATCOR Atmospheric Correction Algorithm

ADC Analog-Digital Converter

AERONET Aerosol Robotic Network

APU Accuracy, Precision and Uncertainty

ASI Italian Space Agency “Agenzia Spaziale Italiana”

ASTM American Society of Testing and Materials

ATBD Algorithm Theoretical Basis Document

AU Astronomical Unit

BMBF Federal Ministry of Education and Research (Germany) “Bundesministerium für Bildung und Forschung”

BOA Bottom-Of-Atmosphere

CCD Charge-Coupled Device

CE Cross-Entropy

CEOS Committee on Earth Observation Satellites

CF Cloud Fraction

CHIME Copernicus Hyperspectral Imaging Mission for the Environment

CH$_2$ Methane

Cinvestav Center for Research and Advanced Studies of the National Polytechnic Institute (Mexico) “Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional”

CMEMS EU Copernicus Marine Environment Monitoring Service

CMIX Cloud Mask Intercomparison eXercise

CNES French National Space Agency “Centre national d’études spatiales”

CNN Convolutional Neural Network

CNN1D 1D - Convolutional Neural Network

CNN2D 2D - Convolutional Neural Network

CNN3D 3D - Convolutional Neural Network

CONV Convolutional

COT Cloud Optical Thickness

CO_2 Carbon Dioxide

CP CANDECOMP/PARAFAC

CPU Central Processing Unit

CTH Cloud Top Height

CUDA Compute Unified Device Architecture

DAAD German Academic Exchange Service “Deutscher Akademischer Austauschdienst”

DB Database

DEM Digital Elevation Model

DESIS DLR Earth Sensing Imaging Spectrometer

DL Deep Learning

DLR German Aerospace Center “Deutsches Zentrum für Luft- und Raumfahrt”

DN Digital Number

DSM Digital Surface Model

ESA	European Space Agency
EM	Electromagnetic
EMIT	Earth Surface Mineral Dust Source Investigation
EnMAP	Environmental Mapping and Analysis Program
EO	Earth Observation
EOC	Earth Observation Center
ERM	Empirical Risk Minimization
EU	European Union
F1	F-1 score
FC	Fully Connected
FE-Net	Feature Extraction Network
FOV	Field Of View
FWHM	Full Width at Half Maximum
GAN	Generative Adversarial Network
GCF	Geometric Cloud Fraction

GEO Geostationary Orbit

GFZ German Research Helmholtz Centre for Geosciences “Helmholtz-GeoForschungsZentrum”

GPU Graphics Processing Unit

GSD Ground Sampling Distance

HISUI Hyperspectral Imager Suite

HPC High Performance Computing

IDL Interactive Data Language

IR Infrared

ISS International Space Station

JAXA Japan Aerospace Exploration Agency

JPL Jet Propulsion Laboratory

L0 Level Zero

L1 Level One

L1B Level-1B

L1C Level-1C

L2 Level Two

L2A Level-2A

Landsat Land Satellite

LAI Leaf Area Index

LEO Low Earth Orbit

LIDAR Light Detection and Ranging

LRZ Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities “Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften”

LWIR Long-wave infrared

LUT Look-Up-Table

MASL Meters Above Sea Level

MBE Mean Bias Error

MCC Matthew’s Correlation Coefficient

nMCC normalized Matthew’s Correlation Coefficient

MODIS Moderate Resolution Imaging Spectroradiometer

MWIR Mid-wave infrared

NASA National Aeronautics and Space Administration (USA)

NDVI Normalized Difference Vegetation Index

NDWI Normalized Difference Water Index

NDMI Normalized Difference Moisture Index

NIR Near-infrared

N_2O Nitrous Oxide

OA Overall Accuracy

OCRA Optical Cloud Recognition Algorithm

OHB Otto Hydraulik Bremen

OPT-MPC Sentinel-2 Optical Mission Performance Cluster

O_2 Molecular Oxygen

O_3 Ozone

P Precision

%pt Percentage Points

PACO Python-based Atmospheric Correction

PCA Principal Component Analysis

PDF Probability Density Function

PINN Physics-Informed Neural Network

PMF Probability Mass Function

PO Polar Orbit

POOL Pooling

PRISMA Hyperspectral Precursor of the Application Mission “Precursore Iperspettrale della Missione Applicativa”

R Recall

RAM Random Access Memory

RCF Radiometric Cloud Fraction

ReLU Rectified Linear Unit

RGB Red-Green-Blue

RMSE Root Mean Squared Error

ROCINN Retrieval of Cloud Information using Neural Networks

RT Radiative Transfer

RTT Radiative Transfer Theory

RTE Radiative Transfer Equation

RTF Radiative Transfer Function

SAR Synthetic Aperture Radar

SBG Surface Biology and Geology

SCL Scene Classification Map

SD Signal-Dependent

Secihti Secretariat of Science, Humanities, Technology and Innovation “Secretaría de Ciencia, Humanidades, Tecnología e Innovación”

SI Signal-Independent

SNR Signal-to-Noise Ratio

SRF Spectral Response Function

SRM Structural Risk Minimization

SSE Sum of Squared Errors

SSO Sun-Synchronous Orbit

SVC Spectra Vista Corporation

SVD Singular Value Decomposition

SWIR Short-wave infrared

TOA Top-Of-Atmosphere

TROPOMI Tropospheric Monitoring Instrument

XAI Explainable Artificial Intelligence

UAV Unmanned Aerial Vehicle

UMAP Uniform Manifold Aproximation and Projection

UML Unified Modeling Language

USGS United States Geological Survey

UTC Coordinated Universal Time

UV Ultraviolet

VIS Visible

VNIR Visible and Near-infrared

H_2O Water

WV Water Vapor

WGCV Working Group on Calibration and Validation

Notation

x, X : a scalar or a pixel.
$\mathbf{x}$: a 1^{st}-order tensor or a vector.
$\mathbf{X}$: a 2^{nd}-order tensor or a matrix.
$\boldsymbol{\mathcal{X}}$: a 3^{rd} or higher-order tensor.
$x_{i,j,...,n}$: the element $(i, j, ..., n)$ of a tensor $\boldsymbol{\mathcal{X}}$.
$\mathbf{x}_i$: the ith column vector of a matrix $\mathbf{X}$.

Chapter 1

Introduction

Nowadays, the remote sensing applications of the Earth's surface have become one of the most important subjects in addressing challenges related to climate change and natural resource management. The recently launched monitoring missions by the most prominent space agencies worldwide demand greater comprehension and assimilation of the huge amount of data generated daily. Different products provide users with various ways to interpret and use the information, depending on the requirements of the application. Therefore, better classification products generally improve the analysis in subsequent processing steps. For instance, masking algorithms provide useful information about the classification of each pixel in the image, allowing users to filter the pixels of interest for analysis. However, many traditional implementations of masking algorithms rely primarily on spectral indices, which are based on a limited number of features and often do not incorporate spatial or temporal context. Although more recent approaches may include neural networks, spatial analysis, or time-series information, there is still a widespread reliance on simpler, threshold-based methods in operational settings. As a result, in certain applications (particularly those focused on small study areas or requiring high precision) scientists may manually adjust the masking products prior to further analysis. However, this additional step increases the required effort and may limit scalability in broader operational contexts.

On the other hand, the growing field of artificial intelligence has demonstrated excellent performance in a wide variety of applications. Given the enormous amount of generated data, remote sensing applications are excellent candidates for modeling solutions with deep neural networks. These networks enable the retrieval of spatial and spectral features with the ability to generalize globally. However, they also present significant challenges, such as the need for a sufficiently diverse set of reference labels for training, as well as robust strategies for validating the resulting masking products. Therefore, recognizing the opportunities to improve classification products using state-of-the-art techniques, this thesis proposes a Deep Learning (DL) approach to enhance the classification products generated by currently implemented masking algorithms.

1.1 Overview

The optical data products of the Earth's surface are essential for environmental monitoring and security, as they capture interactions between solar radiation and matter in the ground and water surfaces. This interaction produces a unique combination of light reflected at different wavelengths for each ground sample, known as the spectral signature. By analyzing these spectral signatures, researchers can gather valuable information for a wide range of applications, including vegetation monitoring, water resource management, climate change assessment, cryosphere monitoring, and wildfire detection, among others [1].

Multiple efforts from space agencies and the private sector have led to the development of missions to monitor Earth, providing access to optical imagery and its associated products. Among the currently operational remote sensing missions [2, 3, 4, 5, 6, 7, 8], the multispectral Sentinel-2 mission [9] stands out as part of the European Union (EU)'s Copernicus program. This Copernicus mission, together with the hyperspectral EnMAP mission [10, 11] from the DLR, will be addressed in this thesis to test the proposed framework.

Remote sensing optical data is captured at various spatial and spectral resolutions, depending on the optical sensor and its specific applications. For instance, multispectral sensors, such as the one onboard Sentinel-2 satellites, have a limited number of selected bands (typically between 5 and 15) with a wide bandwidth, but a Ground Sampling Distance (GSD) of 10-20 m. In contrast, hyperspectral imagery offers finer spectral resolution (between 100 and 250 bands) with narrower bandwidth, acquiring more spectral information to be interpreted by further analysis at a GSD of 30 m [13].

Given the increasing importance of optical remote sensing applications and the potential for better utilization of current distributed products (as well as future missions [14, 15, 16]), an AC process is needed to retrieve an image as if spectroscopy were performed at ground level. Fig. 1.1 shows a schema of this correction. The resulting spectral image is called Bottom-Of-Atmosphere (BOA) reflectance, and it belongs to the often referred Level Two (L2) products (see L2-BOA in Fig. 1.1(e)). The starting point for Atmospheric Correction (AC) is an image composed of the at-sensor Top-Of-Atmosphere (TOA) reflectances, usually distributed in Level One (L1) data (L1-TOA in Fig. 1.1(a)).

During the AC process, it is necessary to identify certain elements captured in the image that correspond to specific atmospheric or Earth's surface features, for instance: clear-sky, water, clouds, among others [12]. For this task, masking algorithms are used, based on thresholds of spectral indices and illumination, to identify patterns of known spectral signatures pixel-by-pixel, generating a mask represented by a binary matrix for each target class (see Fig. 1.2). Another common case is pixel-level classification, often referred to as semantic segmentation or pixel-wise classification, where a unique class is assigned from a defined set of classes for the application.

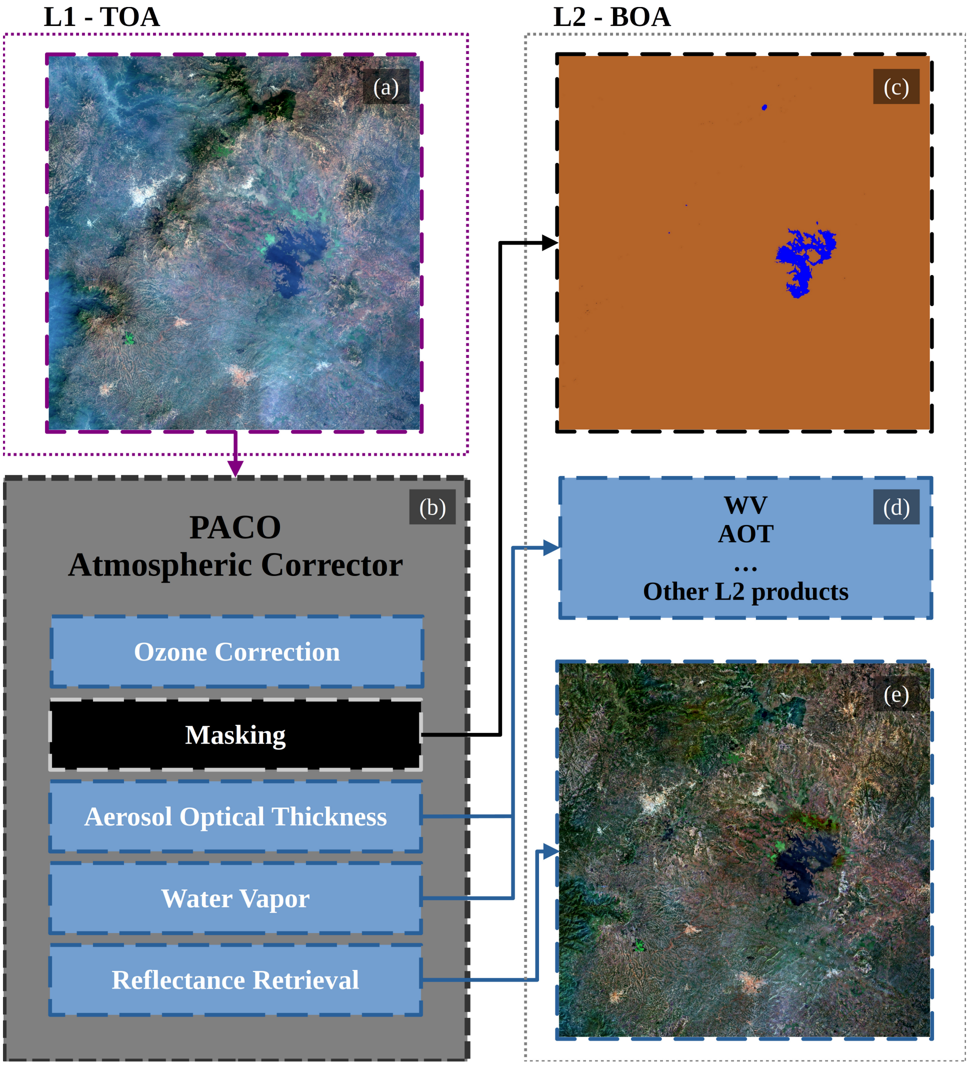


Figure 1.1: Top-Of-Atmosphere (TOA) input image, AC with PACO, and the corresponding products obtained during processing: Bottom-Of-Atmosphere (BOA) image, masks, among others [12]. Image: Sentinel-2, T32NPM, 2016/12/23.

The classification is usually delivered as products after the AC process (L2), either in the form of a set of binary masks (named masks cube, see Fig.1.2c) or as a complete labeled classification map (named pre-classification, see Fig.1.2e), which is obtained by mapping each mask with empirically defined rules. The classification products may also be delivered as a mixture of both for easier use. The final set of classification L2-BOA products is often defined based on the specific requirements of the mission. As a delivered product, this classification provides useful information for a wide range of remote sensing applications. As a result, most of the current remote sensing missions have a state-of-the-

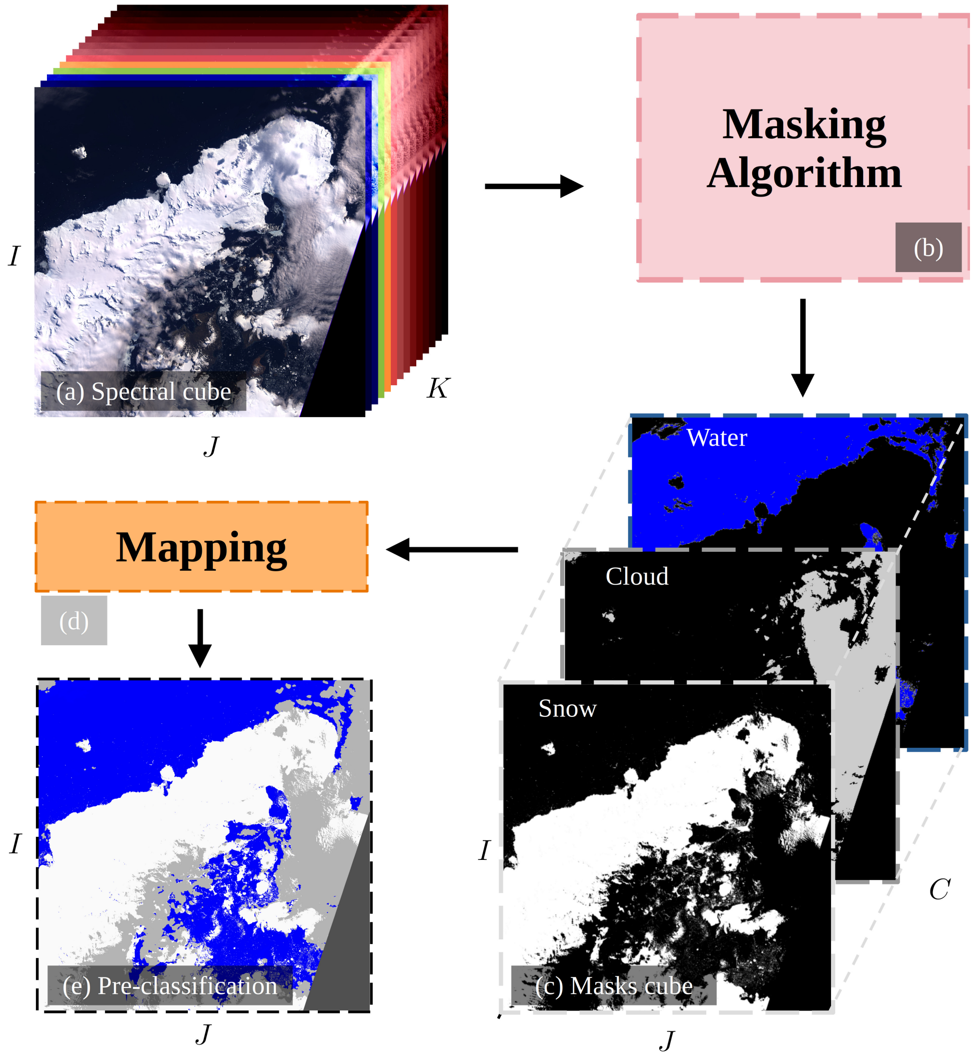


Figure 1.2: Masking algorithm big picture, with a spectral image as an input and the resulting classification map of the same spatial dimensions at the output. Sentinel-2 Image: Antarctic, T21EVK - 2019/02/03.

art implementation of masking algorithms, some of which will be described in Section 2.4.

In the context of the rapidly increasing implementation of machine learning techniques in remote sensing applications, the contribution of this work consist in providing a method for handling high spectral data volumes to train DL classification models with a global-perspective implementation for future optical remote sensing missions. Examples include upcoming hyperspectral missions such as the Copernicus Hyperspectral Imaging Mission for the Environment (CHIME) [14, 15] from the European Space Agency (ESA) and Surface Biology and Geology (SBG) [16] from the National Aeronautics and Space

Administration (USA) (NASA).

The goal of the proposed implementation is to develop a robust methodology to automatically retrieve high-quality classification labels, avoiding the need of manual labeling for creating training datasets. In addition, this work aims to gain insights from testing the models with independent label sources to demonstrate generalization regarding class definitions.

An important aspect to consider is that in this kind of global-scale frameworks, the testing datasets are commonly independently manually labeled, and the model should be trained with reference labels not related with the testing datasets. This demonstrates the generalization ability and robustness of the model. To analyze the bias of each dataset and identify common misclassification problems across all datasets, the resulting confusion matrices and some classifications maps for visualization should be reviewed. This analysis aims to draw conclusions and highlight the improvements needed to create better training datasets.

1.2 Objectives and Contribution

The objectives of this thesis are described in this section, addressing the implementation of a deep learning model for the classification of multispectral and hyperspectral remote sensing imagery.

1.2.1 General Objective

To generate improved classification products of multispectral and hyperspectral remote sensing imagery in the context of an AC processor, by using state-of-the-art DL models with spatial-spectral context feature extraction, training with the state-of-the-art pre-classification products from PACO, which are frequently based on spectral indices, incorporating additional sources of information beyond the spectral signature, such as altitude and illumination conditions of the scene.

1.2.2 Particular Objectives

- To describe how the classification products are improved by training a DL model that extracts spatial and spectral features from the image to identify patterns in previously unused features.

- To include additional data to train the DL model besides the spectral information, such as the Digital Elevation Model (DEM) and the illumination of each pixel. This last one considers the incident angle of light over the scene and the viewing angles of the sensor.

- To evaluate the obtained classifier using independent test sets and assess the results of the classifier.

- To analyze the advantages and weaknesses based on the obtained results and their contribution to classification in remote sensing.

- To identify weaknesses in currently used validation methodologies and to propose bias-free alternatives based on physical parameters retrieved from independent sources of information for classification evaluation.

1.2.3 Contribution

The contributions of this work are presented below, emphasizing the methodological advancements and practical applications derived from the proposed framework. Each point highlights specific aspects that enhance current practices in remote sensing classification, particularly in the context of deep learning and operational processing chains.

- The implementation of DL models for global-scale applications in multispectral and hyperspectral Earth Observation (EO) missions currently in operation is demonstrated, contributing to the state-of-the-art.
- The classification products of the atmospheric processor PACO from DLR are improved through:
    - Performance evaluation of the proposed classifiers.
    - Identification of limitations in current validation methodologies .
    - Proposal of new bias-free validation methods based on physical parameters retrieved from external sources.
- A framework is proposed for the compilation of training datasets and for minimizing manual labeling efforts, enabling the implementation of DL-based masking processors in future missions.
- The methodology is designed to be compatible with real data and can be applied during any operational phase of an remote sensing mission, increasing its practical applicability.

1.3 Thesis structure

This thesis is organized as follows:

- **Chapter 2 – Background and Related Work:** This chapter presents the theoretical foundation relevant to this work, covering key concepts in optical remote sensing and CNNs.
- **Chapter 3 – Methodology:** This chapter details the procedures used to train the model and perform pixel-level classification, including the steps involved in the prediction phase and the validation strategy.
- **Chapter 4 – Results and Discussion:** This chapter presents the experimental results, comparing the proposed method with the baseline pre-classification model from PACO. It also includes a discussion of the observed strengths and limitations of the approach.
- **Chapter 5 – Conclusions:** This chapter summarizes the main findings and insights gained during the development of the thesis and outlines directions for future research.

Chapter 2

Background and Related Work

This work merges two different subjects: optical remote sensing and classifiers based on neural networks, motivated by the significant and widespread impact of CNNs in image processing. DL models have demonstrated excellent adaptability for a wide range of applications that involves high volumes of data. Therefore, a natural step is to extend the current state-of-the-art remote sensing applications using neural networks. However, this is not a plug-and-play task, given the significant challenges associated with global-scale models for Earth's surface optical monitoring missions. In this chapter, the most important concepts for the work are detailed and analyzed.

2.1 Optical Remote Sensing

Remote sensing is defined as the measurement of object properties. It is, therefore, an attempt to measure something *at a distance*, rather than *in situ* [17]. The sensor is not in direct contact with the object of interest and works with propagated signals of some sort, for example, optical, acoustical, or microwaves. In this thesis, the discussion is limited to remote sensing of the Earth's surface using optical signals. This section addresses the basics of light physics and the imaging systems used for remote sensing optical monitoring missions.

2.1.1 Physics of Light

In a nutshell, to measure the properties of an object at a distance, the Electromagnetic (EM) waves emitted by the sun interact with objects or materials on the Earth's surface. The energy is then reflected to the sensor and captured by the imaging system. The resulting signal contains information about the object's physical and chemical properties, which can be decoded through radiometric and geometric analysis. Each physical principle involved in this process (such as photon energy, frequency, wavelength, and the radiometric quantities used to characterize the incoming radiation) is briefly described in Appendix A.1.1 for completeness, as it provides the theoretical foundation upon which remote sensing measurements are based.

2.1.1.1 Electromagnetic Spectrum

In the context of remote sensing, the wavelength of each photon in the EM spectrum determines how it interacts with matter, making it a key variable for interpreting surface and atmospheric features. Light and other forms of EM radiation are commonly described in terms of their wavelengths [18]. In figure 2.1, a representation of the EM spectrum is shown.

For instance, visible light is a type of EM radiation that can be sensed by the human eye. The visible band of the EM spectrum spans the range from approximately 380 nm (violet) to about 750 nm (red), but optical remote sensing is not limited only to the visible range of the EM spectrum; the infrared and the ultraviolet also provide information through their interaction with matter on the Earth's surface.

The following portions of the EM spectrum are widely used in the literature for optical remote sensing [20], with some applications listed alongside their wavelength range definitions:

- Infrared (IR) (0.75 µm - 15 µm) [21]:
    - Thermal infrared(3 µm - 15 µm): Usually divided into Long-wave infrared (LWIR) (8 µm - 15 µm) and Mid-wave infrared (MWIR) (3 µm - 8 µm), both

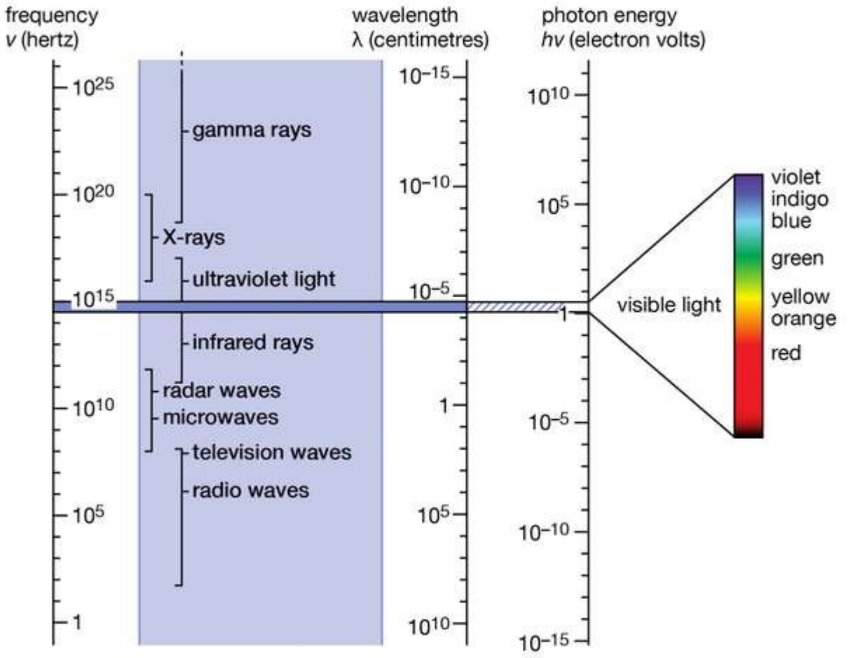


Figure 2.1: Electromagnetic spectrum. Image credit: Encyclopaedia Britannica, 2009 [19].

portions associated with thermal phenomena in the infrared. LWIR is used for night-time imaging and surface temperature mapping, while MWIR is used for detecting heat emissions, wildfire monitoring, and volcanic activity.

  - Short-wave infrared (SWIR) (1.4 µm - 3 µm): This portion is less affected by the atmosphere compared to portions of the EM spectrum with shorter wavelengths and is particularly sensitive to water content in vegetation, soil, and the atmosphere. It is used for agricultural applications, geological mapping, and wildfire damage evaluation.
  - Near-infrared (NIR) (750 nm - 1400 nm): This portion is highly reflected by healthy vegetation. Therefore, it is widely used for vegetation health monitoring, agricultural applications, and water quality assessment. Additionally, some of its bands are used to retrieve atmospheric characteristics, such as Water Vapor (WV) content.

- Visible (VIS) (380 nm - 750 nm) [22]: It is usually categorized into three bands corresponding to the peak sensitivities of the cone cells in the human retina: *Blue* (450 nm – 485 nm), *Green* (500 nm – 565 nm), and *Red* (620 nm – 700 nm). Note that although the visible spectrum nominally extends up to 750 nm, human sensitivity decreases significantly beyond 700 nm. It is used for the “natural” representation of remote sensing imagery, vegetation analysis, land cover classification, and atmospheric analysis such as aerosol concentration estimation. In addition, the Visible and Near-infrared (VNIR) range is widely used in the literature, encompassing the VIS and an adjacent portion of the NIR as a single portion (400 nm - 1100 nm). As an example, the hyperspectral mission EnMAP [23] has two sensors onboard; one capturing a portion of the VNIR (420 nm - 1000 nm) and the second one capturing a portion of the SWIR (900 nm - 2450 nm).

- Ultraviolet (UV) (10 nm - 400 nm) [20]: Since UV radiation is strongly affected by atmospheric components, it is primarily used for atmospheric studies, including ozone estimation and pollution monitoring. In particular, changes in UV backscatter are used to infer aerosol concentrations and trace gases.

Although this classification follows the cited sources, other authors define the wavelength ranges of the EM spectrum differently, even when using the same band names. The rest of the spectrum, such as X-rays, Gamma-rays, and radio waves, are observed using different techniques and are not within the scope of this thesis.

2.1.1.2 Radiative Transfer Theory

Once the solar radiation interacts with the atmosphere and reaches the Earth's surface (readers interested in the interaction of radiation with matter and atmospheric dynamics may refer to the Appendices A.1.1.2 and A.1.1.3), it begins its return path toward space. At this stage, the portion of energy reflected or emitted by the surface is detected by a sensor onboard a satellite or airborne platform. In functional terms, a sensor is a device that captures radiometric information across specific spectral bands and converts it into digital signals for further processing. This mechanism forms the basis of remote sensing, enabling the observation and characterization of surface properties from a distance. The relevant spectral ranges used in this work are summarized in Subsection 2.1.1.1, while specific details about the optical sensors involved are discussed in Section 2.1.2.

However, the sensor does not capture a single light beam in isolation; instead, it receives a mixture of radiative components resulting from multiple reflections and scattering events occurring in the atmosphere and at the Earth's surface. These interactions are described and modeled by Radiative Transfer Theory (RTT). For the purposes of this thesis, only a brief and intuitive overview of RTT is included, as the detailed modeling is already handled by the atmospheric correction processor, whose output serves as the input to the algorithm developed in this study.

“The Radiative Transfer Equation (RTE) for a homogeneous surface under clear sky conditions can be formulated as”[24]:

$$L = L_p + \frac{\tau \left(E_{dir} \cos\theta_s + E_{dif}\right) \rho / \pi}{1 - \rho \cdot s} \tag{2.1}$$

where: L = At sensor radiance,
L_p = Path radiance (reflected from the atmosphere),
τ = Ground-to-sensor transmittance,
E_{dir}, E_{dif} = Direct and diffuse solar flux on the ground,
θ_s = Solar zenith angle,
ρ = Surface BOA reflectance,
s = Spherical albedo of the atmosphere.

Since L is the radiance measured by remote sensing sensors, it is possible to solve the RTE for ρ using different approaches, being one of the most widely used being the numerical approximation through radiative transfer models [25]. These models estimate each variable in Equation (2.1) based on known atmospheric conditions (e.g., aerosol content, water vapor, ozone), as well as observational geometry, including solar zenith and sensor viewing angles. The use of Look-Up-Tables (LUTs) as precomputed solutions for a wide variety of atmospheric conditions is common during the AC process, discussed further in Subsection 2.2.3.2.

These models estimate each variable in Equation (2.1) based on

From the obtained reflectance ρ (referred to as BOA reflectance), insights are gained about the interaction of incoming light with matter, excluding atmospheric influence. This allows for consistent analysis over time under variable atmospheric conditions.

2.1.1.3 Spectral Signature

The wide variety of matter on the Earth's surface and constituents of the atmosphere, in addition to all possible illumination conditions and light scattering, spans an infinite set of possible combinations of captured radiant flux by optical sensors. The generic term for this is *spectrum*, and in plural, *spectra*.

The unique combination of reflected flux for each wavelength that characterizes the elements present in a captured scene is known as the *spectral signature*. Even though there are many sources contributing to variability, in general terms, spectra of similar matter are highly correlated. Therefore, the common spectral segments between samples are considered to represent the spectral signature. An example of three different types of matter and their associated spectral signatures is shown in Figure 2.2.

The spectral signature is the main characteristic used for classification, given that allows us to find patterns between samples and define common classes based on them, i.e. soil, water, snow, cloud, vegetation, etc.

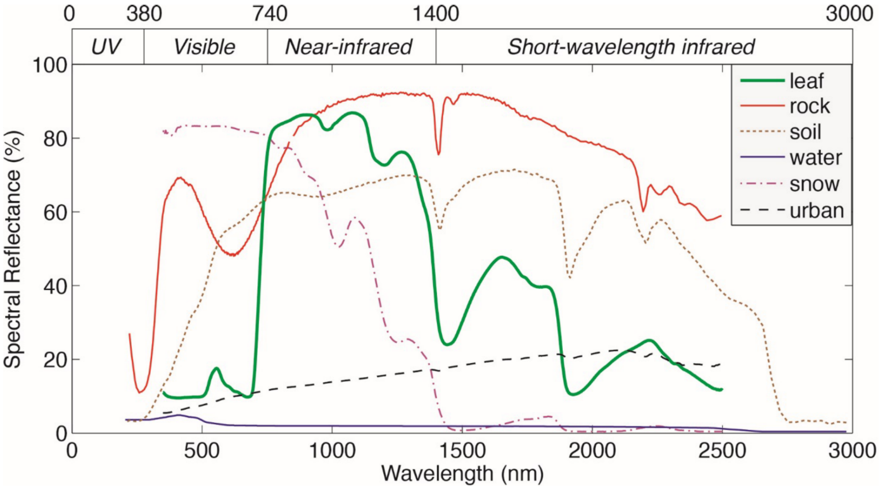


Figure 2.2: Different characteristic spectral signatures of leaf, rock, soil, water, snow and urban areas. "Reflectance data are derived from the United States Geological Survey (USGS) Digital Spectral Library. The laboratory measurements represent samples of an oak leaf from Colorado (leaf), Aventurine quartz from India (rock), montmorillonite and illite from Virginia (soil), seawater from the Pacific Ocean (water), fresh snow from Colorado (snow), and black road asphalt from Colorado (urban)". Source: EnMAP science plan [26].

2.1.2 Optical Sensors

In this section, the operating principles of an optical imaging system are intuitively described. By briefly describing the capturing process, important concepts are addressed, including different types of resolution, the covered area, details about each specific mounting platform, and the noise processes associated with electronics and optical system construction.

All these aspects are crucial as they define the characteristics and quality of the output spectral image, ultimately determining its suitability for a given application. This is particularly relevant in the context of this thesis, which involves models applied across multiple sensors and validated using reference datasets with varying spatial and spectral resolutions.

2.1.2.1 Operating Principles

Similar to the generic EM sensor concept described in Appendix A.1.2.1, a spectral sensor can be understood as a collection of detectors, each capturing the energy reflected from a scene within a specific portion of the EM spectrum. These discrete portions, commonly referred to as bands or spectral channels, together form the sensor's spectral coverage. For instance, the sensors onboard EnMAP span a spectral range from 418.2 to 2445.5 nm, divided into 224 bands [11].

Extending the definition of a digital image in AppendixA.1.2.1[27], a spectral image is a three-dimensional function $f(x, y, z)$, where x and y are *spatial* (plane) coordinates, and z is the *spectral* coordinate. The amplitude of f at any triplet of coordinates (x, y, z) represents the radiance measured by the sensor at that spatial location and wavelength. A spectral cube is illustrated in Figure2.3.

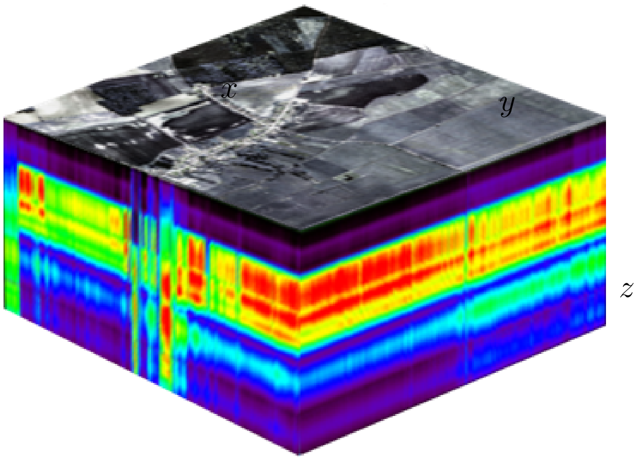


Figure 2.3: Spectral cube from EnMAP mission. Image credit: German Research Helmholtz Centre for Geosciences “Helmholtz-GeoForschungsZentrum” (GFZ)

Using the notation from [28] and the representation of spectral cubes with tensors, where a is a scalar, $\mathbf{a}$ is a vector, $\mathbf{A}$ is a matrix, and $\boldsymbol{\mathcal{A}}$ is a 3rd- or higher-order tensor, a spectral image of an observed scene $\boldsymbol{\mathcal{X}}$ is represented as a 3rd-order tensor:

$$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}, \tag{2.2}$$

where I (rows) and J (columns) are the spatial dimensions of the image, and K is the number of spectral bands. Then, the element $x_{i,j,k}$ is the reflectance value at position (i, j) in band k, such that:

$$\{\forall i, j, k \in \mathbb{Z} \mid (1 \leq i \leq I) \wedge (1 \leq j \leq J) \wedge (1 \leq k \leq K)\}, \tag{2.3}$$

and the vector containing the spectral signature of a pixel is represented by $x_{i,j,:}$. Each ordered pair (i, j) represents coordinates of a specific location on the Earth's surface in the observed scene.

2.1.2.2 At Sensor Reflectance

Remote sensing sensors do not measure radiance values directly. Instead, they record a digitized signal called the Digital Numbers (DNs), whose precision depends on the sensor's radiometric resolution, defined by the number of bits used to encode each pixel. This bit depth determines the sensitivity to radiance variations (lower resolutions can cause quantization noise and loss of detail, see Appendix A.1.2.3 for further explanation and visual examples).

L_k is the measured radiance for band k, defined by [12, 29]:

$$L_{i,j,k} = c_0^k + c_1^k \cdot DN_{i,j,k}, \tag{2.4}$$

with DN as the scaled radiance (named Digital Number) and the radiometric offset c_0^k and gain c_1^k obtained during radiometric calibration.

The *reflectance* of a pixel refers to TOA reflectance (also named at-sensor reflectance) represented by the Greek letter $\rho_{i,j,k}^{\text{TOA}}$, and computed for each of the k bands. It is defined in [12, 29] as follows:

$$\rho_{i,j,k}^{\text{TOA}} = \frac{\pi \cdot d^2 \cdot L_{i,j,k}}{E_k \cdot cos(\theta_s)}, \tag{2.5}$$

where d represents the Earth-Sun distance in astronomical units, E_k is the extraterrestrial solar irradiance for band k, and θ_s denotes the solar zenith angle at the time of acquisition. Most of the classification and masking algorithm have as input the TOA reflectance.

2.1.2.3 Multi/Hyperspectral Imaging

The response of each spectral band corresponds to a specific range of wavelengths where the sensor is sensitive, typically characterized by its central wavelength and the bandwidth of its Spectral Response Function (SRF). These concepts, along with related topics such as responsivity, spectral sampling, and band definition, are discussed in more detail in Appendix A.1.2.2.

The spectral resolution of optical remote-sensing missions can be grouped, in order of increasing band count, usually into five classes: *panchromatic* (1 broad band), *RGB* (3 bands), *multispectral* (≈5–15 bands), *superspectral* (≈10–100 bands), *hyperspectral* (≈100–250 bands), and *ultraspectral* (hundreds to thousands of ultra-narrow bands).

In this work, we focus exclusively on the multispectral and hyperspectral sensors. Figure shows 2.4, a simplified visualization to compare the spectral features categories.

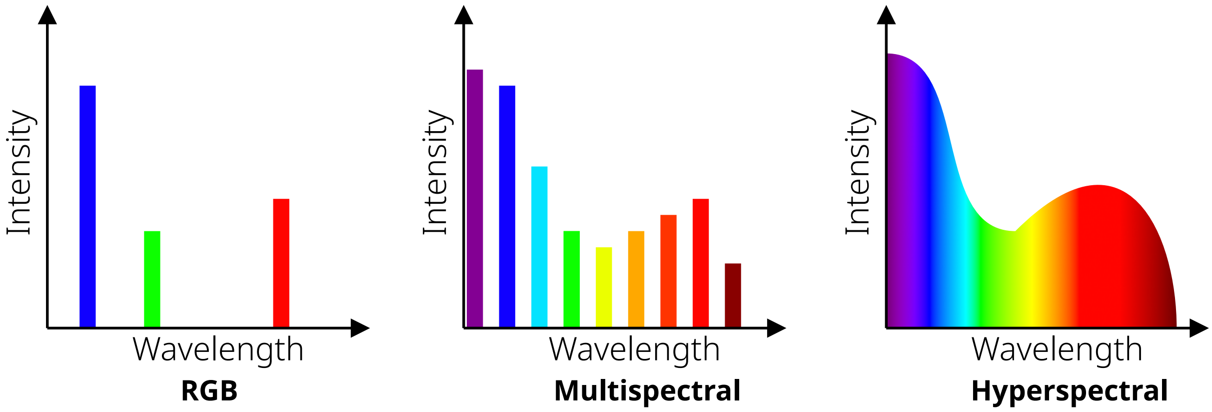


Figure 2.4: A simplified visualization of the spectral features of Red-Green-Blue (RGB), multispectral, and hyperspectral imagery for comparison, with different number of spectral features.

- Multispectral sensors have a limited number of selected bands (typically between 5 and 15) with a wide bandwidth, covering the most relevant wavelengths in the VNIR-SWIR range to retrieve physical properties of the Earth's surface and atmosphere. Examples of currently operational missions include Sentinel-2 [9], Land Satellite (Landsat) [8, 30, 31], Moderate Resolution Imaging Spectroradiometer (MODIS) [32, 33], among others.
- Hyperspectral sensors offer finer spectral resolution (between 100 and 250 bands), with Full Width at Half Maximum (FWHM) values below than or equal to 10 nm, acquiring more spectral information. Examples of currently operational missions are EnMAP [10, 11] and DLR Earth Sensing Imaging Spectrometer (DESIS) [5, 34] from DLR, Earth Surface Mineral Dust Source Investigation (EMIT) [35, 36] from

NASA Jet Propulsion Laboratory (JPL) Hyperspectral Precursor of the Application Mission “Precursore Iperspettrale della Missione Applicativa” (PRISMA) [4] from Italian Space Agency “Agenzia Spaziale Italiana” (ASI) and Hyperspectral Imager Suite (HISUI) [37] from Japan Aerospace Exploration Agency (JAXA).

The tendency is that multispectral sensors tend to have a smaller GSD (better spatial resolution, explained below) with broader FWHM at specific wavelengths of interest and a shorter revisit time, allowing for constant monitoring. On the other hand, hyperspectral imagery usually offers uniformly distributed bands over the EM spectrum with narrower FWHM, allowing a “continuous” interpretation of the spectra at the expense of increased storage and processing requirements due to the extensive amount of spectral features.

Spatial resolution refers to the level of detail a sensor can capture on the Earth’s surface, typically defined by the GSD. Most optical EO missions use push-broom scanners, which acquire image rows as the satellite moves forward, with each pixel representing a footprint on the ground. The width of the swath and sensor geometry determine the GSD, while motion and integration time define pixel dimensions. A more detailed explanation of this acquisition process is provided in Appendix A.1.2.4.

2.2 Remote Sensing Missions

The increasing interest in remote sensing data from space agencies, governments, and the private sector has led to the launch of a steadily growing number of satellites dedicated to EO each year. This growth parallels the rising interest from the scientific community, which aims to develop algorithms for processing images and deriving products for EO applications.

Sensors used in optical remote sensing can be deployed on a variety of platforms, including satellites, aircraft, drones, and ground-based instruments. Each platform type influences the image resolution, coverage, and frequency of observations, making them suitable for different monitoring purposes. In-situ measurements from ground-based spectrometers are particularly important for validating the accuracy of satellite-derived reflectance data, as shown in Figure 2.5. A detailed description of the platforms and their characteristics is provided in Appendix A.1.2.5.

Figure 2.5: In-situ measurements for PACO-Level-2A (L2A) validation campaign of Sentinel-2 and EnMAP missions. Klosterwiese, Fürstenfeldbruck, Bavaria.

The current status of optical remote sensing research looks promising, with exciting new missions upcoming for hyperspectral data, such as CHIME [14, 15] from the ESA, and the SBG mission from NASA. In Figure 2.6, a plot from [38] provides an overview of current and future spectroscopy missions.

A compromise between spectral, spatial, and temporal resolution is present in optical remote sensing missions. However, as storage capacity, computational resources, and sensor construction and capabilities increase, newer missions are surpassing the current state. In the following subsections, the main characteristics and derived products of the remote sensing missions used throughout this thesis are described in detail. These include the Sentinel-2 mission for multispectral imagery (Section 2.2.1), the EnMAP mission for hyperspectral data (Section 2.2.2), and the Sentinel-5P mission with its TROPOMI sensor for atmospheric monitoring (Section 2.2.6.1). Additionally, the PACO processor

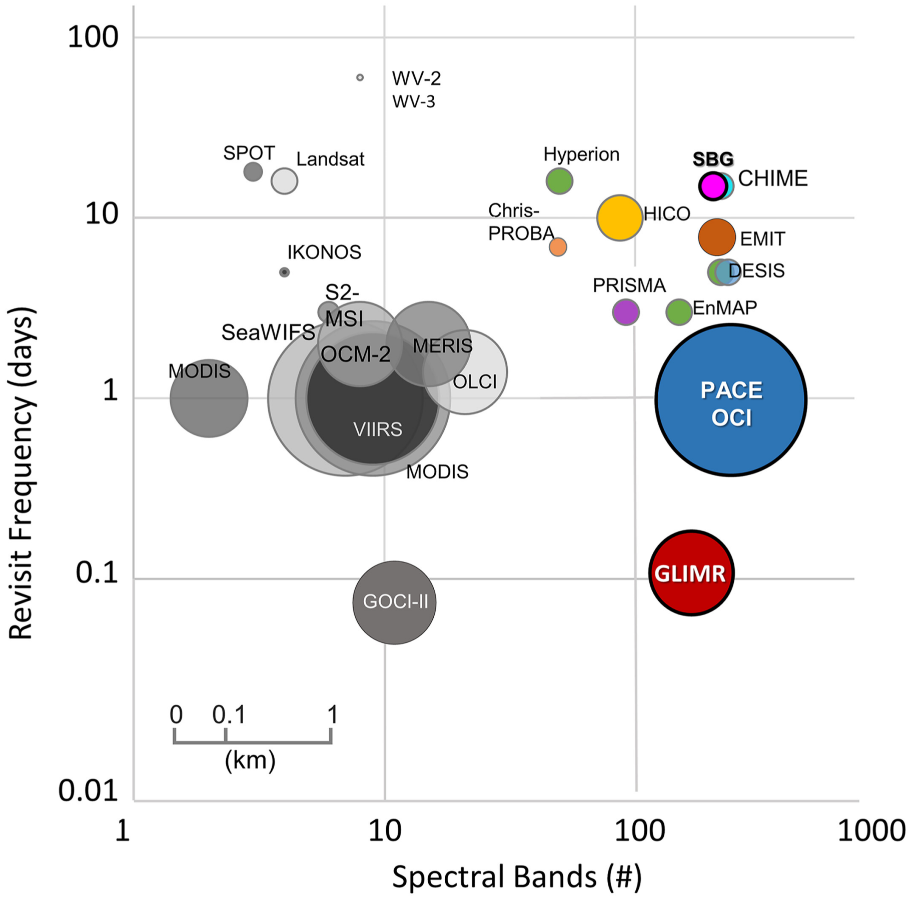
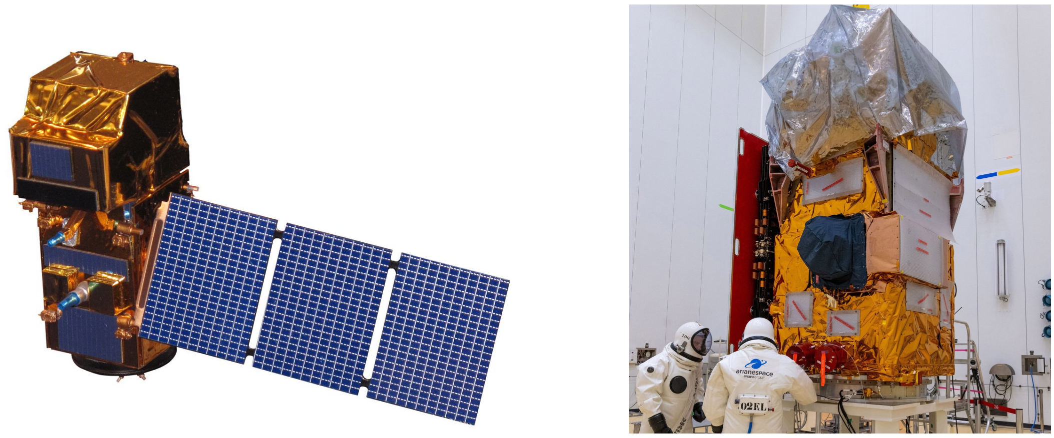


Figure 2.6: Overview of operational and future multispectral and hyperspectral missions. Circle size indicates GSD. Gray circles denote multispectral, colored denote hyperspectral missions. Plot credit: [38].

(Section 2.2.4) and related processing levels (Section 3.4) are presented, as they play a crucial role in generating products derived from atmospheric correction used in this work. These sections establish the foundational understanding necessary for interpreting the input data and the subsequent processing stages analyzed in this thesis.

2.2.1 Multispectral: Sentinel-2

The first satellite of the Sentinel-2 mission was launched in 2015 and it belongs to the Copernicus Program of the ESA. Nowadays, it consists of a constellation of three satellites, all equipped with sensors of identical characteristics (Sentinel-2A, -2B, and -2C) flying at a mean altitude of 786 km in an Sun-Synchronous Orbit (SSO) with 98.62° of inclination, capturing multispectral imagery with a high revisit frequency of 10 days at the Equator for each spacecraft, which is reduced to 5 days when two satellites are available [39]. The recent launch of Sentinel-2C in September 2024 was intended to replace the Sentinel-2A spacecraft at the end of its lifetime, but recently, the ESA made the decision to keep the spacecraft, extending the operational phase exceptionally for one year [40]. In Figure 2.7, the schematic of the Sentinel-2 spacecraft is shown.



(a) Spacecraft. Image credit: ESA.

(b) Spacecraft before launch. Image credit: ESA-French National Space Agency "Centre national d'études spatiales" (CNES)-ARIANESPACE/Optique vidéo du CSG–S. Martin.

Figure 2.7: Sentinel-2.

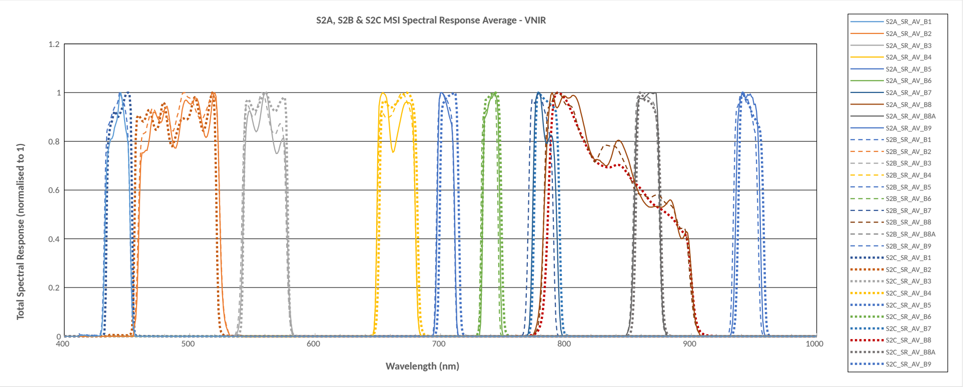
"Each of the satellites in the Sentinel-2 mission carries a single payload: the optical multispectral instrument that samples 13 spectral bands: four bands at 10 m, six bands at 20 m, and three bands at 60 m spatial resolution. The orbital swath width is 290 km" [39]. In Table 2.1, the description of the bands' characteristics is shown.

As mentioned in subsection A.1.2.2, although the spectral response of each band is assumed to be Gaussian, the shape varies by band for the multispectral sensor onboard each spacecraft of the Sentinel-2 mission. In Figures 2.8a and 2.8b are displayed the SRFs for each band of the VNIR and SWIR sensors, respectively, onboard each satellite (A, B and C) of the Sentinel-2 mission.

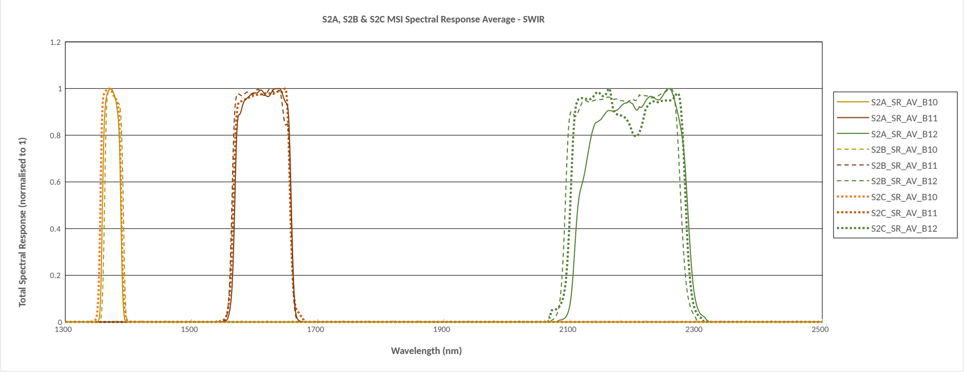
Spatial Resolution (m)	**Band #**	**Central Wavelength** (nm)	**FWHM** (nm)	**Purpose**
60 m	1	443	20	Aerosol scattering
	9	945	20	Water vapour absorption
	10	1375	30	Detection of thin cirrus
20 m	5	705	15	Position of red edge
	6	740	15	Retrieval of aerosol load
	7	783	20	Leaf Area Index (LAI)
	8a	865	20	Sensitive to total chlorophyll, biomass
	11	1610	90	Snow/ice/cloud separation
	12	2190	180	Soil, live and dead biomass
10 m	2	490	65	Aerosol scattering
	3	560	35	Green peak, sensitive chlorophyll
	4	665	30	Maximum chlorophyll absorption
	8	842	105	LAI

Table 2.1: Band characteristics of Sentinel-2 sensors. Each satellite slightly varies in the FWHM and central wavelength due to differences in construction. Source: Copernicus Program [41].

Figure 2.8: Spectral response of Sentinel-2 mission sensors with three satellites in the constellation: Sentinel-2A,-2B and -2C. Plot credit: Copernicus Program [41].

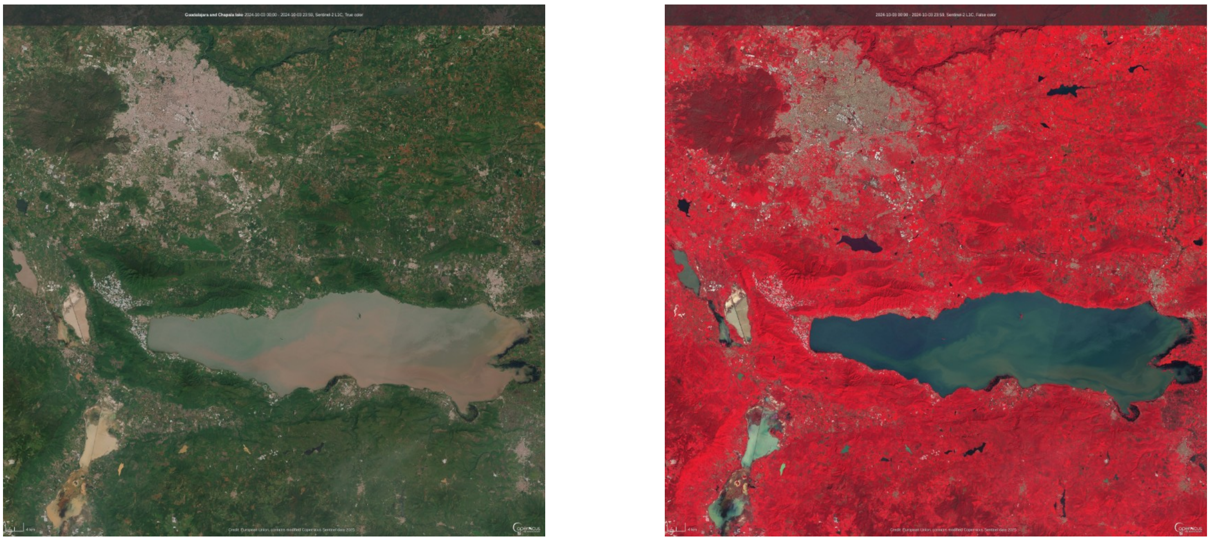


(a) VNIR spectral response functions.



(b) SWIR spectral response functions.

Since its launch, multispectral imagery and derived products from the Sentinel-2 mission have been publicly provided to the scientific community, leading to a significant number of studies utilizing the data for new algorithms and applications [42]. The Level-1C (L1C) product contains ortho-images representing TOA reflectance, radiometrically corrected and projected onto a cartographic grid (UTM/WGS84). In Figure 2.9 an example of RGB and False color composites of a Sentinel-2 multispectral image.



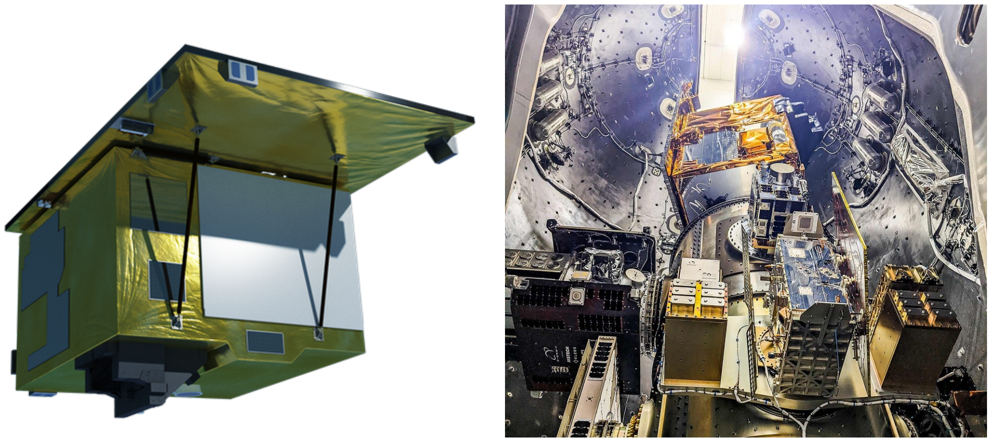
(a) RGB True Color composite based on bands B4 (Red), B3 (Green), and B2 (Blue).

(b) False color composite based on bands B8 (NIR), B4 (Red), and B3 (Green) to RGB components directly. This composite is used to assess plant density and health.

Figure 2.9: Sentinel-2 multispectral image products. Guadalajara city and Chapala lake in the image, captured on 2024/10/03 over Jalisco, Mexico. Image credit: Copernicus data space.

2.2.2 Hyperspectral: EnMAP

The Environmental Mapping and Analysis Program (EnMAP) is a German hyperspectral satellite mission that monitors and characterizes Earth's environment on a global scale. It was launched in April 2022. The project is managed by DLR Space Agency in cooperation with GFZ (Scientific principal investigator) and Otto Hydraulik Bremen (OHB) (Space segment) [43]. The DLR's ground segment is in charge of the operations, payload, and processing. It consists of two hyperspectral sensors (for VNIR and SWIR) flying at a mean altitude of 650 km in an SSO with 97.96° of inclination, capturing hyperspectral imagery with a swath width of 30 km and GSD of 30 m × 30 m close to the equator, with a target revisit time of 27 days (viewing zenith angle $\leq 5°$). The sensor can be tilted to point to a specified location, reducing the revisit time to 4 days at the expense of larger viewing zenith angles of capture (viewing zenith angle $\leq 30°$) [11]. In Figure 2.10, the schematic of the EnMAP spacecraft is shown.



(a) Spacecraft. Image credit: Earth Observation Center (EOC) of DLR [44].

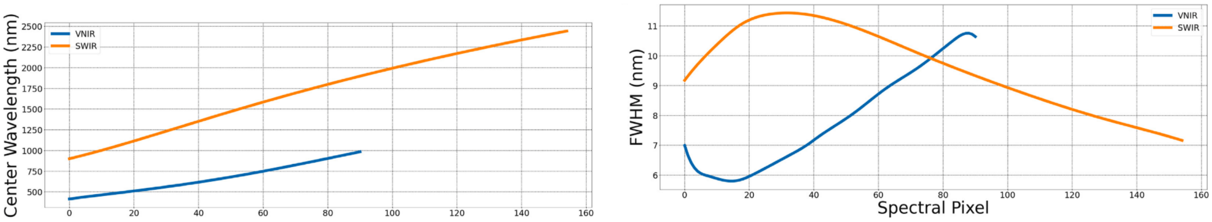
(b) Spacecraft before launch on the Falcon 9 rocket. Image credit: SpaceX.

Figure 2.10: EnMAP.

Onboard the EnMAP spacecraft, two hyperspectral sensors capture the VNIR (420 nm - 1000 nm) with a spectral sampling interval of approximately 6.5 nm in 91 bands, and the SWIR (900 nm - 2450 nm) with a spectral sampling interval of approximately 10 nm in 155 bands, with a total of 246 bands (with an overlap between 900 nm - 1000 nm) [11].

The FWHM varies depending on the captured wavelength, and the spectral response functions are characterized as Gaussian, as explained in Appendix A.1.2.2. In Figure 2.11, plots display the center wavelength and its FWHM for both sensors [11].

"The primary goal of the EnMAP mission is to provide accurate and diagnostic information for research and applications in fields such as agriculture and forestry, geology



(a) Center wavelength.

(b) FWHM.

Figure 2.11: Center wavelength and FWHM for each spectral band of VNIR and SWIR sensors. Plot credit: Storch et al. [11].

and soils, urban areas, coastal and inland waters" [44]. The EnMAP data is publicly available since its launch for the researchers community. The L1C product contains geo-referenced hyperspectral images representing TOA (or at-sensor) radiances in units of [W $\mathrm{m}^{-2}\,\mathrm{sr}^{-1}\,\mathrm{nm}^{-1}$], radiometrically corrected and geo-referenced in (UTM/WGS84). In Figure 2.12 an example of RGB composites of EnMAP hyperspectral on two examples of urban areas.

(a) Guadalajara city (top) and Cajititlán Lagoon (bottom right).

(b) San Marcos Lagoon (left) and Chapala Lake (right).

Figure 2.12: RGB True Color composites based on bands at 850 nm (Red), 670 nm (Green), and 550 nm (Blue), of EnMAP. Both images were captured on 2024/01/05 over Jalisco, Mexico. Image credit: EnMAP DLR.

2.2.3 Processing Levels

Remote sensing missions continuously capture new spectral data, requiring a processing pipeline to make the products available to end users for further analysis. This processing is performed by the "Ground Segment," which is responsible for designing and implementing algorithms for radiometric and geometric correction in the Level One (L1) level of processing, and AC in the Level Two (L2). In Figure 2.13, a simplified diagram of the processing pipeline performed by the ground segment of a remote sensing mission is shown. Each processing level is described in the following subsections.

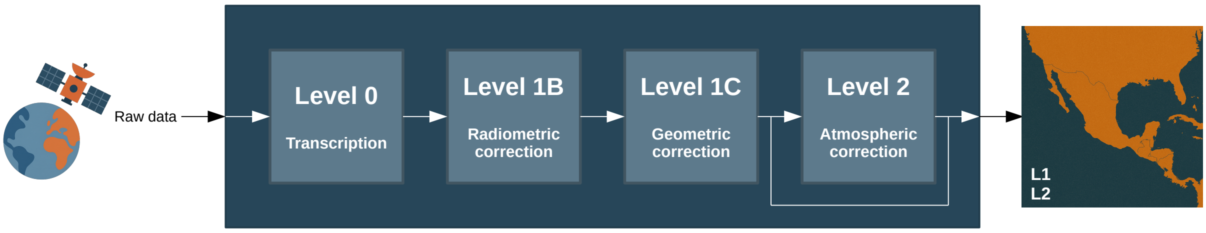


Figure 2.13: Ground segment processing pipeline of a remote sensing mission.

The pipeline's different levels of processing vary in their definitions and names across remote sensing missions. The generic example presented in this subsection is based on EnMAP's ground segment processing [45].

2.2.3.1 Level 0 (Raw Data) and Level 1 (Top-of-Atmosphere)

After the capture performed by the sensor, the raw data is transmitted from the spacecraft to a terrestrial base, which stores it and makes it available for further processing in the pipeline. The data is composed of an Earth data take tile, together with auxiliary information, such as orbit and altitude information, capture time, calibration tables, and other metadata. This product is known as Level Zero (L0).

The L1 is divided into two processing levels (in common for EnMAP and Sentinel-2 missions):

- The Level-1B (L1B) [46] takes the L0 product as input, where systematic and radiometric corrections are performed from the DNs to obtain calibrated TOA or at-sensor radiances (in $\mathrm{W\,m^{-2}\,sr^{-1}\,nm^{-1}}$ or normalized to the solar irradiance). The processing includes various algorithms related with radiometric calibration using measured dark signals; a trimming process; and sensor non-linearity correction, among other processes and quality check routines.
- The L1C [47] takes the L1B product as input, where geometric corrections related to satellite motion and terrain distortion are performed. Then, the image is resampled to a specific grid (orthorectification) and georeferenced, making it ready for the users and further levels of processing in TOA radiances, like L2 for AC.

2.2.3.2 Level 2 (Bottom-of-Atmosphere)

The Atmospheric Correction (AC) takes as input the radiometrically corrected TOA radiances, orthorectified, and geocoded L1C products, together with the cropped and coregistered DEM, matching the image dimensions and geolocation. In addition, a collection of databases provides information about solar irradiance and Radiative Transfer (RT), along with external auxiliary information that varies with the processor. In the case of EnMAP, they consist in atmospheric properties, like the air temperature and the total column of Ozone (O_3) concentration [24].

Several atmospheric correction processors have been developed for different remote sensing missions. For Sentinel-2, Sen2Cor [48] is the official processor developed by ESA. The ATCOR algorithm [49] has been widely used across multiple missions and forms the basis of PACO [12], a Python-based implementation developed by DLR. The ISOFIT framework [50] offers a physically rigorous, modular approach to atmospheric correction using optimal estimation. For hyperspectral data from PRISMA, the official PRISMA L2 processor [51] provides operational surface reflectance products. These processors differ in complexity, radiative transfer modeling, and the types of auxiliary data used but all aim to retrieve accurate surface reflectance under variable atmospheric conditions.

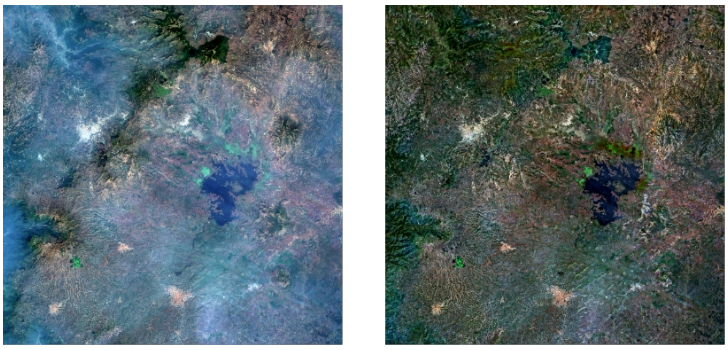


Figure 2.14: TOA (left) and BOA (right) side-by-side-comparison. Image: Sentinel-2, T32NPM, 2016/12/23.

The main output products of L2 are the Bottom-Of-Atmosphere (BOA) reflectances, representing surface reflectance values as if the measurements were taken directly at the Earth's surface, without the influence of atmospheric effects. These products are retrieved under constantly varying atmospheric conditions, which introduce absorption and scattering. By correcting for these effects, L2 products enhance the comparability of measurements across time and sensors, enabling more consistent scientific analyses, broader data usage, and statistically robust insights. In addition, products generated during the AC are also included in the output products, such as the pre-classification masks of various classes of interest, e.g., land, water, cloud, cirrus, haze, cloud and topographic shadows, snow/ice, no data (background), or L1 data quality information, like dead pixels [12, 24]. Figure 2.14 shows an example of an output image in BOA reflectance, within the L2 products.

As a final remark, this thesis is developed in the context of this level of processing, performing classification tasks using as input the products generated at the early stages of L2 processing, more specifically concerning the pre-classification, aiming to improve the current delivered products to be used for later processing steps and users.

2.2.4 PACO

The Python-based Atmospheric Correction (PACO) [12] is a proprietary atmospheric correction software developed by DLR for L2 processing. It is a Python implementation of the established ATCOR algorithm, originally written in Interactive Data Language (IDL), and is used to perform atmospheric correction of multispectral and hyperspectral imagery. Currently the supported sensors by PACO include Sentinel-2, Landsat-8 (multispectral), and DESIS and EnMAP (hyperspectral). However, the inherent modularity of PACO allows the core to be reused, enabling the easy implementation of additional spaceborne remote sensors.

As a L2 processor, the main input consists of the TOA radiances along with the associated metadata, but information from external databases is also used during processing (elevation, pre-computed Radiative Transfer Functions (RTFs), solar irradiance spectra, among others). The primary output of PACO is the surface reflectance image at BOA level. In addition, PACO exports several intermediate products generated throughout the atmospheric correction process, such as cloud and shadow masks, snow detection, illumination geometry parameters (e.g., sun and view angles), and elevation information. These products are particularly relevant in this thesis, as some of them were selected as inputs to the classification models. A detailed overview of PACO's outputs is provided in Table 2.2.

Product Name	**Label**	**Units**
Masks	hcw	
Dense-dark Vegetation mask	ddv	
Visibility index	visindex	
Aerosol Optical Thickness	aot	1000 * [unit-less]
Water vapor column	wv	1000 * [cm]
Illumination	ilu	
Quicklook	ql	
BOA surface reflectance	atm	100 * [%]

Table 2.2: Intermediate and final products resulting from PACO software execution. Table source: De Los Reyes, et al. [12].

From all these products, only those generated prior to the final atmospheric correction were selected; masks, illumination and DEM (this last one is a external input of the atmospheric correction). described in Subsection 2.2.5.2. The remaining products fall outside the scope of this work, as the aim is to perform classification at early stages of the atmospheric correction process.

Within PACO, masks are internally a multi-label classification of the pixels, describing the type of matter on the Earth's surface or in the atmosphere that blocks the view (completely or partially). It includes 21 different masks as described in Table 2.3, obtained by applying logical combinations of thresholds over spectral indices and specific decision rules. Detailed definitions of the classes can be found in [12], and additional information about mask computation is provided in Subsection 2.2.5.1, while complete definitions are available in [24].

Class #	**Class Name**	**Description**
0	Background	Pixels flagged as background or with a data value of zero.
1	Shadows	Spectrally detected shadows, including possible topographic effects.
2–4	Cirrus over Water	Thin (2), medium (3), and thick (4) cirrus clouds identified over water.
5	Land	Clear land pixels not flagged by other mask criteria.
6	Saturated	Pixels likely affected by sensor saturation.
7	Snow/Ice	Pixels classified as snow or ice.
8–10	Cirrus over Land	Thin (8), medium (9), and thick (10) cirrus clouds identified over land.
11, 13	Haze	Haze detected over land (11) and over water (13).
15–16	Clouds	Clouds detected over land (15) and water (16).
17	Water	Water pixels under clear atmospheric conditions (no haze or clouds).
18–19	Cirrus Cloud	Cirrus (18) and thick cirrus (19) where surface type (land/water) is uncertain.
20	Bright	High-reflectance pixels not otherwise flagged as cloud or haze.
21	Topographic Shadows	Shadows cast by terrain, calculated using DEM and solar geometry.

Table 2.3: Masks labels, name, and description. Information extracted from: [12]

The binary cube formed by the internally computed masks is represented as:

$$\boldsymbol{\mathcal{M}} \in \{0,1\}^{I,J,\text{Classes}}, \tag{2.6}$$

where a pixel can belong to one or more classes. From a pixel perspective, the vector is one-hot encoded, indicating with a high value the class or classes to which it belongs, described as follows:

$$\mathbf{x}_{i,j,:} \in \{0,1\}^{\text{Classes}}. \tag{2.7}$$

Then, a mapping based on empirical rules is performed from the masks cube to a classification map with a unique integer label, named "hcw" in Table 2.3. This mapping is defined as follows:

$$f_{\text{mapping}} : \boldsymbol{\mathcal{M}} \rightarrow \mathbf{M} \in \mathbb{Z}^{I,J}. \tag{2.8}$$

2.2.5 Products

Among a wide variety of derived products from remote sensing missions, this subsection includes a brief description of the products used in this thesis. These are obtained from L1C products processed at the early stages of L2 processing, or (in the case of the DEM) obtained from external sources.

2.2.5.1 Masking

Masking (also called semantic segmentation) is based on identifying pixels that belong to a class based on their spectral features. The product is a *mask* (in the form of a binary matrix) with the same dimensions as the image, containing *True* values at the locations (coordinates) where a pixel of the class was detected. Usually masking algorithms rely on spectral indices, which are simple mathematical combinations (usually ratios or differences) of the reflectances at different wavelengths (bands) of a multispectral or hyperspectral image.

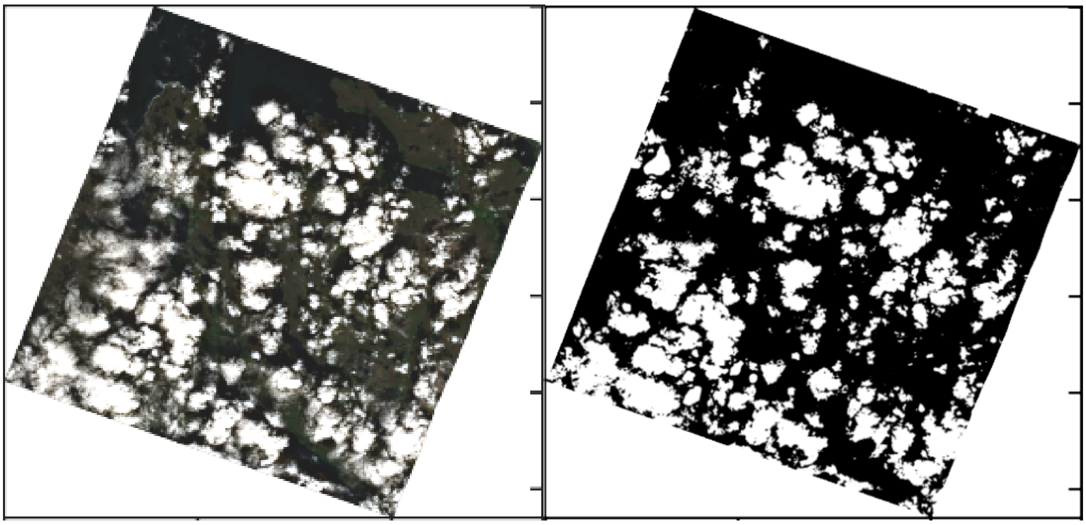
Masking algorithms leverage spectral indices (see Appendix A.3) by using a defined set of rules composed of empirical thresholds applied to one or multiple spectral indices, ratios, or simply reflectance ranges. These rules often include multiple conditions evaluated together using logical expressions, along with other sources of information such as altitude, slope, or season, among others.

The masking process is performed class by class, obtaining a binary mask for each one with the same spatial dimensions as the spectral image, resulting in a *mask cube* used for internal processing during AC. As an example, the case of *cloud over land* detection in PACO for the L2 processing of EnMAP [24] is addressed. The cloud mask is defined as follows:

$$\rho^*(\text{Blue}) > T_c \text{ and } \rho^*(\text{Red}) > 0.15 \text{ and } \frac{\rho^*(\text{NIR})}{\rho^*(\text{Red})} < 2 \text{ and } \rho^*(\text{NIR}) > 0.8 \cdot \rho^*(\text{Red}) \text{ and }$$
$$\frac{\rho^*(\text{NIR})}{\rho^*(\text{SWIR1})} > 1 \text{ and NDSI} < 0.7 \text{ and } DN(\text{blue}) > T_{\text{saturation}} \quad (2.9)$$

where: p^* = TOA reflectance,
T_c = Cloud threshold, default $T_c = 0.25$,
$T_{saturation}$ = Saturation DN, default $T_{\text{saturation}} = 14746$.

Figure 2.15 shows an example of cloud-over-land detection using the PACO masking algorithm.



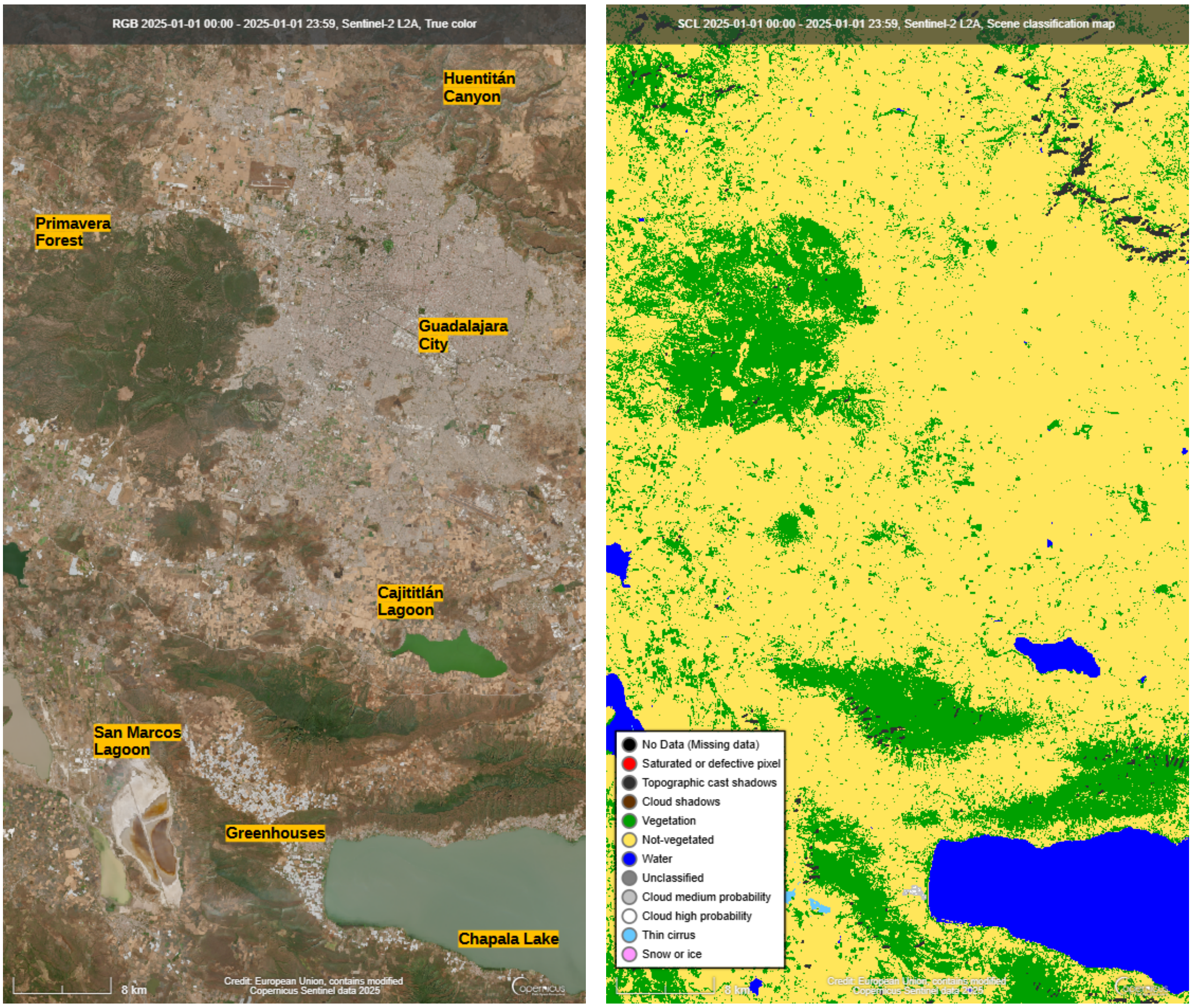
(a) RGB.

(b) Cloud Mask.

Figure 2.15: True color and cloud mask computed using equation (2.9). Image: EnMAP, DT0000032132_005, captured on 2023/08/10.

One of the commonly distributed product in L2 is a classification map, which combines each mask by mapping a unique class label. As an example, Figure 2.16 shows the Scene Classification Map (SCL) processed by Sen2Cor [52], the atmospheric processor of the Sentinel-2 mission.

Some of the classes are not obtained from spectral information. For example, the *topographic cast shadows* are obtained by computing the cast shadows using the DEM (see Subsection 2.2.5.2) and the sun angles, present at the top right (where the Huentitán Canyon is located, with an average depth of 600 m) in Figure 2.16.



(a) RGB.

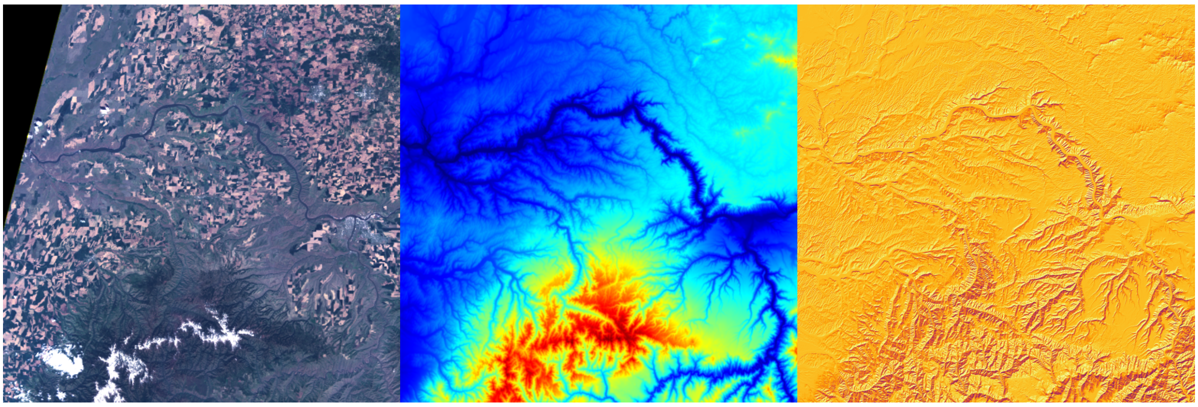
(b) SCL.

Figure 2.16: True color and SCL from Sen2Cor. Image: Sentinel-2, over Guadalajara and surrounding areas, captured on 2025/01/01. Source: Copernicus Data Workspace.

2.2.5.2 Digital Elevation Model and Illumination Map

In addition to spectral data, a Digital Elevation Model (DEM) provides information about the altitude of every ground mapped pixel on Earth, expressed in Meters Above Sea Level (MASL). The DEM includes only the elevation of the bare Earth. Usually, a DEM is obtained by filtering a Digital Surface Model (DSM), while the DSM also includes buildings, trees, infrastructure, and any object on the Earth's surface.

These are created using a wide variety of technologies, such as Synthetic Aperture Radar (SAR), Light Detection and Ranging (LIDAR), and stereo images to estimate terrain elevation relative to a previously measured reference point. The Copernicus program provides the GLO-30, a DSM with worldwide coverage at 30 m resolution, acquired by the TanDEM-X mission between 2011 and 2015 [53]. In Figure 2.17b, an example of the DEM GLO-30 is shown, cropped and resampled to match a Sentinel-2 scene.



(a) RGB. (b) DEM. (c) Illumination map.

Figure 2.17: (a) RGB composite for visual reference . (b) DEM from GLO-30, resampled to match the Sentinel-2 scene, with elevation ranging from deep blue (lowest, riverbed) to dark red (highest, mountaintop). (c) Illumination map derived from solar geometry, highlighting cast shadows caused by terrain. Scene: Sentinel-2 tile T11TMM over Rimrock, US, captured on 2018/05/12.

In Figure 2.17c, an example of an illumination map is shown, computed after deriving slope and aspect from the DEM in Figure 2.17b, together with the solar zenith and azimuth angles. This map is used during L2 processing to detect cast (topographic) shadows in the masking stage, and it also supports the AC process, as steep terrain—under the assumption of Lambertian reflectance—can produce over-corrected (i.e., artificially bright) surface reflectance values. In the image, cast shadows are clearly visible in areas with strong topographic variation, particularly in the mountainous region at the bottom and along the river valley traversing the scene.

2.2.6 Atmosphere Monitoring

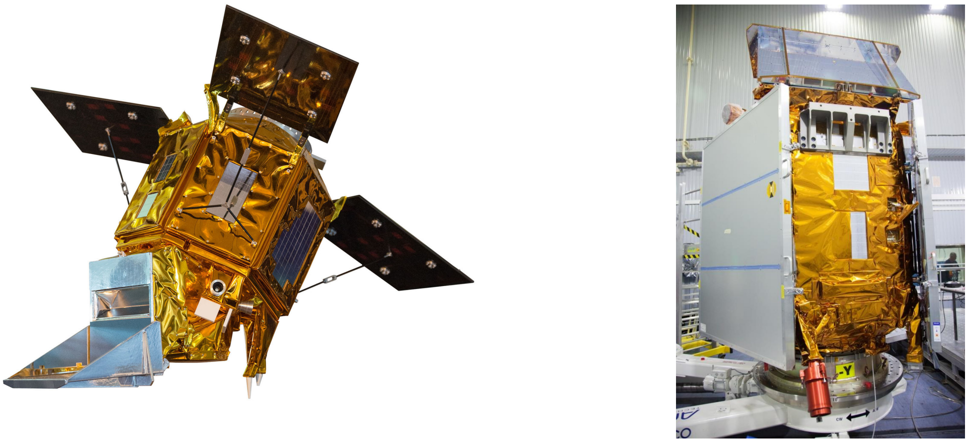
In contrast to remote sensing missions designed for observing the Earth's surface, some missions are specifically developed for atmospheric monitoring, providing valuable products for meteorology, air quality assessment, UV radiation monitoring, and climate change studies.

These missions offer global coverage on a daily basis, at the expense of lower spatial resolution, which remains suitable for monitoring atmospheric phenomena.

2.2.6.1 Sentinel-5P

Sentinel-5 Precursor is the first mission of the Copernicus Program dedicated to monitoring the atmosphere with high spatio-temporal resolution. Its applications include air quality monitoring, measurements of ozone and UV radiation, and climate monitoring and forecasting [54].

The payload is the Tropospheric Monitoring Instrument (TROPOMI), which consists of four spectrometers (each with two bands, totaling eight) for UV (270 nm – 320 nm), VIS (320 nm – 490 nm), NIR (710 nm – 775 nm), and SWIR (2305 nm – 2385 nm), flying at a mean altitude of 824 km in an SSO (near-polar orbit) with a 98.7° inclination, capturing spectral radiance with a swath width of 2600 km and a GSD of 5.5 km × 3.5 km, allowing full daily surface coverage [54]. In Figure 2.18, the Sentinel-5P spacecraft is shown.



(a) Spacecraft. Image credit: EU Copernicus Marine Environment Monitoring Service (CMEMS).

(b) Spacecraft before launch. Image credit: ESA.

Figure 2.18: Sentinel-5P.

2.2.6.2 Sentinel-5P Products

In addition to the spectral data, a compilation of useful derived products is provided within the mission. In Table 2.4, a summary is provided of the derived products obtained from the mission processed by different organizations.

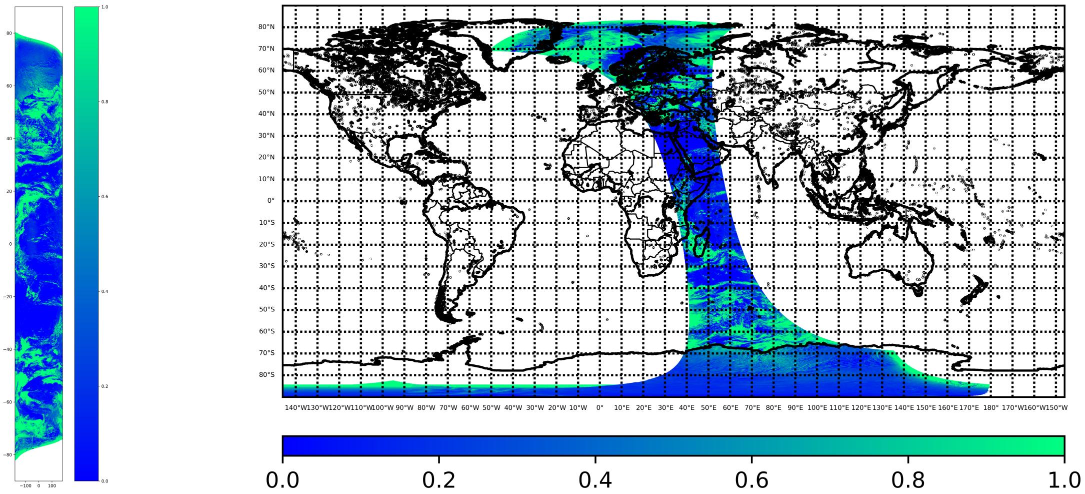
Product	**Identifier**	**Institution**
Cloud	L2__CLOUD__	DLR
NPP-VIIRS Clouds	L2__NP_BDx	RAL
HCHO	L2__HCHO__	BIRA/DLR
SO_2	L2__SO2__	BIRA/DLR
O_3 Total Column	L2__O3___	BIRA/DLR
O_3 Tropospheric Column	L2__O3_TCL	IUP/DLR
Aerosol layer height	L2__AER_LH	KNMI
Ultra violet aerosol index	L2__AER_AI	KNMI
O_3 Full Profile	L2__O3_PR	KNMI
NO_2	L2__NO2__	KNMI
CO	L2__CO__	SRON/KNMI
CH_4	L2__CH4__	SRON/KNMI

Table 2.4: Sentine-5P products and their developing institutions. Source Sentinel-5P/TROPOMI L2 Products User Manual [55].

An extensive collection of products for trace gas concentration retrieval and cloud characterization is provided. Given the objectives of this thesis, the interest relies on the mentioned cloud products, described in [56], and not on the spectral data, used as reference for cloud classification assessment for the hyperspectral mission EnMAP.

Belonging to the cloud products, the Cloud Fraction (CF) is obtained using the Optical Cloud Recognition Algorithm (OCRA) to retrieve the percentage of the pixel covered by clouds. The cloud fraction retrieval is computed radiometrically, considering the UV and VIS bands and analyzing the influence of clouds by comparing changes in reflectance with a database of cloud-free background reflectance [56]. Using the cloud fraction as an input, the algorithm of Retrieval of Cloud Information using Neural Networks (ROCINN), obtains the Cloud Top Height (CTH) and Cloud Optical Thickness (COT), also named albedo. “The algorithm is based on the comparison of measured and simulated satellite sun-normalized radiances in and near the Molecular Oxygen (O_2) A-band”[56]. In Figure 2.19, examples of Sentinel-5P CF visualizations are shown.

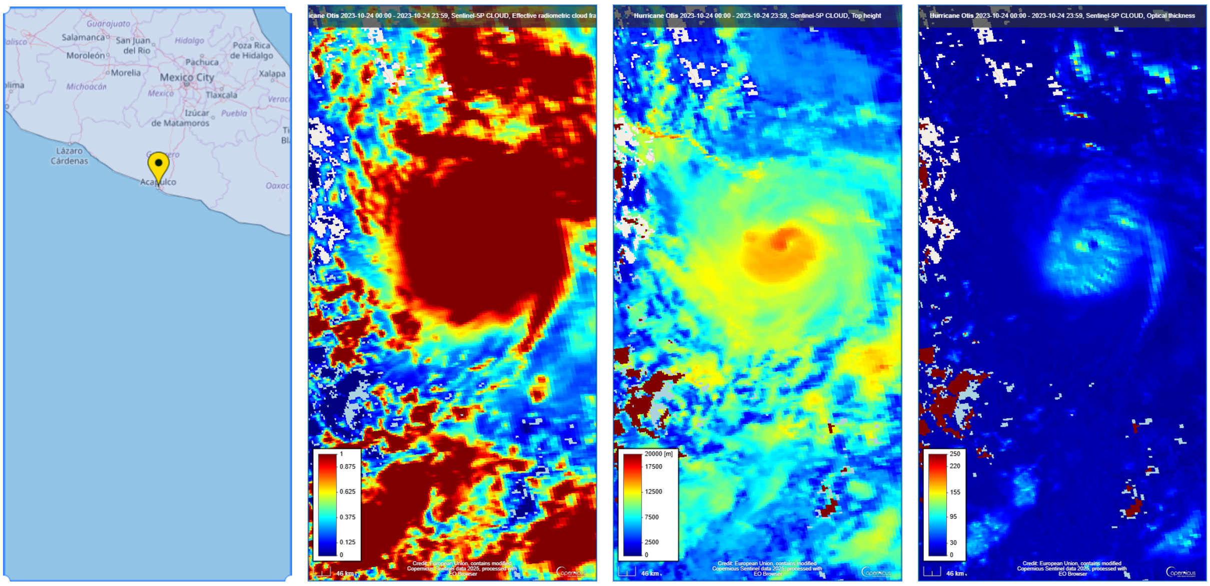
These products are used to monitor and analyze large-scale atmospheric phenomena such as tropical cyclones, hurricanes, and severe storm systems by providing critical input for weather forecasting models, early warning systems, and climate research. In Figure 2.20, the mentioned products are shown highlighting *Hurricane Otis*, a category 5 major hurricane which impacted Acapulco, Mexico, causing significant damage and loss of life [57].



(a) CF in its matrix form.

(b) CF map in an equidistant cylindrical projection.

Figure 2.19: CF retrieved from Sentine-5P data. Image captured on 2024/03/05. Capture start and end (in Coordinated Universal Time (UTC)): 09:31:31 - 11:13:02.



(a) Reference Map.

(b) CF.

(c) CTH.

(d) COT.

Figure 2.20: Hurricane Otis. Sentinel-5P Image captured on 2023/10/24 over Acapulco, Guerrero, México.

2.3 Classifiers Based on Neural Networks

“Artificial neural networks are popular machine learning techniques that simulate the mechanism of learning in biological organisms” [58]. Neural networks are composed of *neurons*, which are computational units connected to one another through *weights* (analogous to synaptic connections in biological organisms). Each input to a neuron is affected by the weight of the connection, which influences the function computed at that unit [58]. “An artificial neural network computes a function of the inputs by propagating the computed values from the input neurons to the output neuron(s) and using the weight as intermediate parameters. Learning occurs by changing the weights connecting the neurons” [58].

Like biological organism, the learning is based in external stimuli. For the case of supervised learning of a classifier, the neural network training need to feed it with *training data* composed of two components:

$$(\mathbf{X}, \mathbf{y}) \quad \mathbf{X} \in \mathbb{R}^{N \times D}; \mathbf{y} \in \mathbb{Z}^{N} \tag{2.10}$$

where: $\mathbf{X}$ = The data that characterize each sample,
$\mathbf{y}$ = The reference label indicating the class of the sample.

$\mathbf{X}$ is the input data of the model with N samples and D dimensions (features), and $\mathbf{y}$ the desired output.

During training, the function (usually called neural network model) is fed with the data and then the output (prediction) is evaluated, denoted as $\hat{\mathbf{y}}$. If the prediction is wrong, the neural network receives *feedback* to adjust the weights. Therefore, the error is propagated, modifying the model's weights in order to better fit the data and produce improved predictions in future iterations [58]. These iterations are called *epochs*.

For instance, in the implementation case of spectral imagery classification, a neural network can be trained to predict the class a pixel belongs to. The training data is composed of the spectral signature (previously mentioned in Subsection 2.1.1.3) of a set of pixels and their reference labels (e.g., vegetation, land, water...). During training, the model extracts the features that make the spectral signature unique and allow it to differentiate each sample among the set of possible classes.

The correctness of the prediction depends on how well the training data represents the entire universe of possible samples and how capable the model is at extracting characteristic features, among other factors. If a neural network is fed a completely different sample (an unknown spectral signature), the prediction will be based on the current knowledge learned from the training dataset, and the predicted class will be the one closest in terms of distance defined in the model's latent space.

2.3.1 Spectral Data

This subsection addresses all the important topics to consider in per-pixel classification of multispectral and hyperspectral imagery. In this thesis, the concept of *spectral data* is used to refer to all the information contained in all wavelengths and neighboring pixels that define a pixel, together with additional data retrieved from external sources and temporal series.

2.3.1.1 Spectral Data Space

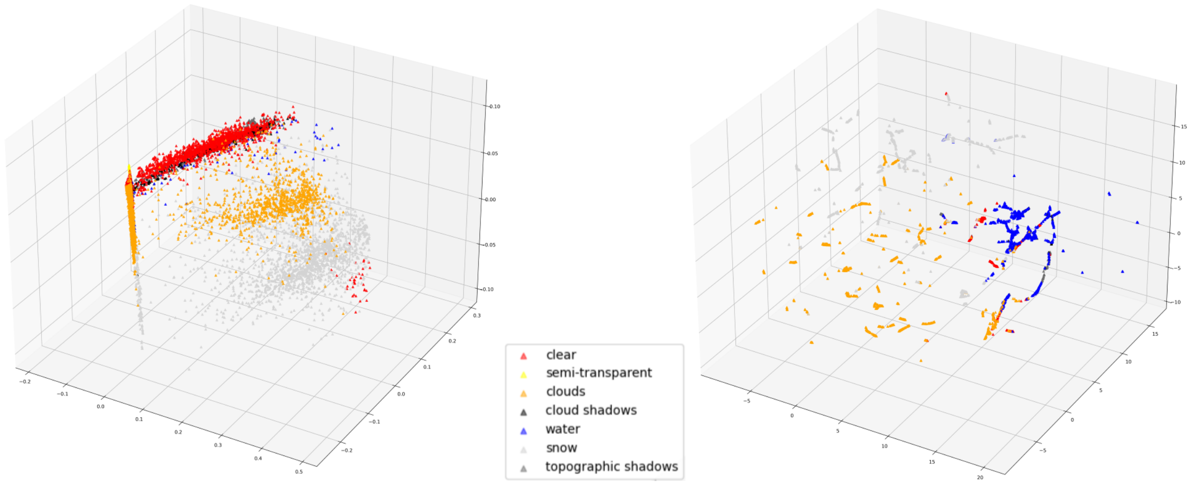
Considering a single location on the Earth's surface with coordinates (i, j), let S be a *stochastic process* that defines interactions between the sunlight and the observed material, and let T be a *time series* of observations of S, considering that S evolves (or changes) over time according to an associated Probability Density Function (PDF). The changes in the PDF are caused by many factors related to natural phenomena and human intervention.

Then, when an optical sensor captures an image that contains the location (i, j), the associated pixel at the specified location is an observation z_t at some timestamp $t \in T$, corresponding to the capture time. Therefore, the observation z_t corresponds to a *realization* of a random variable with Probability Mass Function (PMF) $p(z_t)$. Similarly, the observations z_t captured at times t that belong to a time series are described by an N-dimensional random variable $(z_1, z_2, \ldots, z_N)$ with the associated PMF $p(z_1, z_2, \ldots, z_N)$ [59].

Therefore, all the measured variables related to the changes in the PDF of the stochastic process S contribute to the characterization of the pixel capturing the location (i, j), e.g., the captured spectrum, viewing angles, sun angles, atmospheric conditions, surface temperature, biome, season, among many others. In addition, given that remote sensing data tend to be spatially correlated (locally), the spatial information of neighboring pixels helps to characterize the pixel of interest (only if the GSD is small enough), e.g., a pixel in the middle of a cloud, a water body, or an agricultural crop. The temporal series also add valuable information, but these are out of the scope of this thesis.

According to the presented definition of a realization, each pixel represents a data point (a vector) in the data space (a vector space), which is defined by all possible locations (i, j) on Earth's surface. Typically, each sample is composed of F features representing the spectrum, which explains the intensity values measured at each wavelength (additional features may also be included). The number of features is known as the *dimensionality* of the data.

To visualize how the data points are distributed in the data space, it is common to use dimensionality reduction techniques such as Principal Component Analysis (PCA)[60] or Uniform Manifold Aproximation and Projection (UMAP)[61], allowing the data to be represented in a new space with smaller dimensionality (e.g., three or two dimensions for visualization in a 3D or 2D plot, respectively). An example is shown in Figure 2.21, where even when using the same data, completely different data space representations can be obtained depending on the dimensionality reduction technique applied.



(a) PCA. (b) (c) UMAP.

Figure 2.21: 3D scatter plot of two different data spaces obtained using dimensionality reduction techniques, showing the class of each pixel in different colors. Data corresponds to a Sentinel-2 image (with 13 spectral features) over Antarctica: T21EVK, captured on 2019/02/03.

Analogously to the presented 3D representation, in the original F-dimensional space, the data points are distributed in a similar manner within this high-dimensional space. Depending on the case, the dimensionality can be reduced without losing variance in the data, e.g., when a variable is fixed or proportional to another variable.

2.3.1.2 Curse of Dimensionality

Although high dimensionality allows for better characterization of data due to the increased amount of information, handling the resulting high-dimensional space introduces new challenges, as intuitions from 2D or 3D spaces do not extrapolate to higher-dimensional ones. The set of problems that arise when processing a large number of features is known as the *curse of dimensionality* [62].

The first challenge is that *distance* behaves differently in high-dimensional spaces. Compared to low-dimensional spaces (2D or 3D), where distance is typically measured using the *Euclidean distance*, increasing the number of features causes differences between data points to become less distinguishable. For example, in hyperspectral imagery (~ 200 bands), a homogeneous region (e.g., an agricultural crop) will have many similar feature values. A local anomaly (e.g., a small portion of the crop under stress due to a plague) may result in slight variations in a single or a few adjacent bands. From the perspective of

Euclidean distance, all data points may appear equally far apart, as these small variations are negligible compared to the overall feature set [62].

The second challenge is that, as the number of features increases, data tend to become sparse in the high-dimensional space. Each additional dimension expands the space exponentially, requiring a proportionally larger number of samples to maintain adequate coverage of the high dimensional data space. For neural network-based classifiers, the unavailability of such large labeled datasets makes overfitting a key concern when working with high-dimensional inputs [63].

The third challenge is the increased likelihood of introducing noise. Not every added feature contributes useful information; some may be irrelevant to the target patterns and only introduce noise. If a classifier learns from such irrelevant bands, its performance may be decreased, as those features can mask small but meaningful variations for pattern detection [64].

Nevertheless, the challenges posed by the curse of dimensionality can be mitigated using various strategies. One approach is band selection, which identifies the most informative features through importance analysis [65]. Another common method involves dimensionality reduction techniques such as PCA [60], Singular Value Decomposition (SVD) [66], or tensor algebra approaches like the Tucker or CANDECOMP/PARAFAC (CP) decompositions [28]. Additionally, neural networks can be trained with a focus on avoiding overfitting, employing methods such as regularization or early stopping [67].

This thesis adopts early stopping as the main strategy, since conventional dimensionality reduction techniques project data into a new vector space derived from the training distribution, limiting their applicability to unseen scenes in a global-scale context. Although band selection based on feature importance is a promising alternative, the full spectral signature was preserved to retain subtle patterns essential for distinguishing spectrally similar classes, such as clouds and snow. Incorporating regularization (particularly guided by a physically meaningful parameter) is considered a valuable next step, to be combined with early stopping in future developments.

2.3.1.3 Data Normalization

The data features must have the same value range to ensure equal weighting at the neural network input. Common normalization techniques include Z-score normalization (also known as the standard score) and Min-Max normalization, as described in equations (2.11) and (2.12), respectively. These transformations are applied to each dimension d of the input data.

$$z_{i,j,d} = \frac{x_{i,j,d} - \mu_d}{\sigma_d} \tag{2.11}$$

where μ_d is the mean and σ_d is the standard deviation of feature d. After normalization, the transformed features have $\mu_d = 0$ and $\sigma_d = 1$.

$$x'_{i,j,d} = \frac{x_{i,j,d} - x_d^{\min}}{x_d^{\max} - x_d^{\min}} \in [0, 1] \tag{2.12}$$

where $x_d^{\max}$ and $x_d^{\min}$ are the maximum and minimum values of feature d, respectively. The transformed feature values are constrained to the range $[0, 1]$.

2.3.2 Single-Layer Neural Network and Basic Concepts

This section explains some basic concepts of neural networks using a minimal architecture as an example. The theory is then extended to multi-neuron and multi-layer networks.

Also known as a *perceptron*, this simple network connects all inputs directly to a single output through a linear function. It consists of one input layer and one output node, as illustrated in Figure 2.22 [58].

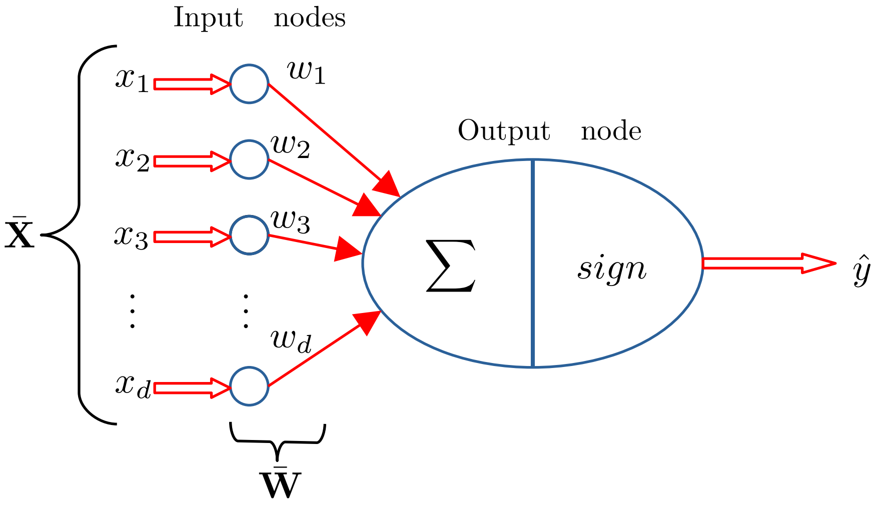


Figure 2.22: Perceptron without bias.

Considering the tuple of the form $\left(\bar{\mathbf{X}}, y\right)$, where each $\bar{\mathbf{X}} = [x_1, \cdots, x_d]$ contains the d feature variables, and $y \in \{-1, +1\}$ contains the *reference* binary class variable (often called *ground truth*, but in this thesis avoided because noise in labels is expected). "Reference" refers to the fact that this information is provided as training data, and the goal is to predict the class variable for cases in which it is not known (or observed). The input layer contains d nodes that transmit the d features via edges with weights $\bar{\mathbf{W}} = [w_1, \cdots, w_d]$ to an output node. This input layer does not perform any computation on its own [58]. The following linear function is then computed at the output node:

$$\bar{\mathbf{W}} \cdot \bar{\mathbf{X}} = \sum_{i=1}^{d} w_i x_i \tag{2.13}$$

Finally, the sign of the result in Equation 2.13 is used to predict the class of $\bar{\mathbf{X}}$. Therefore, the prediction $\hat{y}$ is computed as:

$$\hat{y} = \text{sign}\left\{\bar{\mathbf{W}} \cdot \bar{\mathbf{X}}\right\} = \text{sign}\left\{\sum_{j=1}^{d} w_j x_j\right\} \tag{2.14}$$

The linear function in Equation 2.13 can be improved by adding a *bias neuron*, which prevents it from being constrained to pass through the origin, thereby improving classification performance. This is achieved by adding a neuron that always transmits a value of 1 to the output node, allowing the entire linear activation function to shift without changing its shape [58]. This bias is illustrated in Figure 2.23.

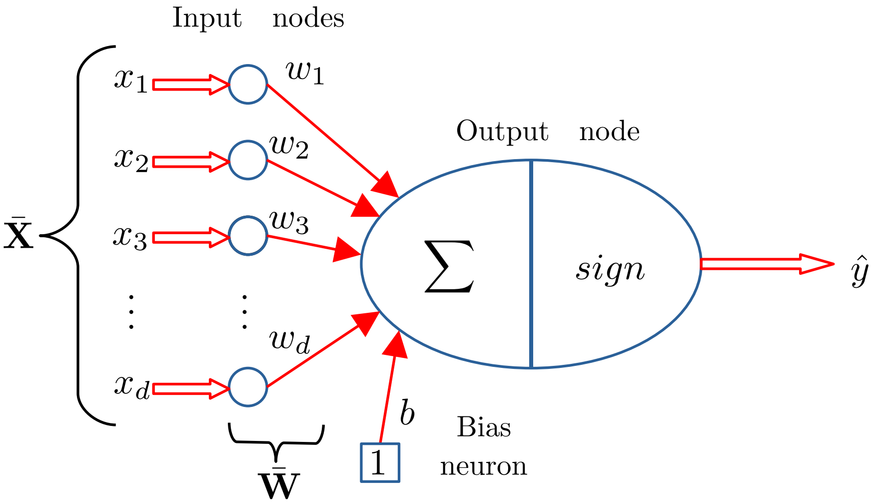


Figure 2.23: Perceptron with bias.

2.3.2.1 Activation functions

There are nonlinear activation functions, which become essential when using multi-layered architectures. Let Φ denote the activation function:

$$\hat{y} = \Phi\left(\bar{\mathbf{W}} \cdot \bar{\mathbf{X}}\right) \tag{2.15}$$

The most basic activation function, which provides no nonlinearity, is the identity or linear activation [58]:

$$\Phi(v) = v \tag{2.16}$$

Linear activation functions are often used in the output node when the target is a real-valued variable, in tasks known as *regression*. One of the most popular activation functions is Rectified Linear Unit (ReLU), defined as follows:

$$\Phi(v) = \max(0, v) \tag{2.17}$$

Classical activation functions for *classification* include the sign (Equation 2.18), sigmoid (Equation 2.19), hyperbolic tangent (Equation 2.20), and hard hyperbolic tangent (Equation 2.21) functions [58]:

$$\Phi\left(v\right) = \operatorname{sign}\left(v\right) \tag{2.18}$$

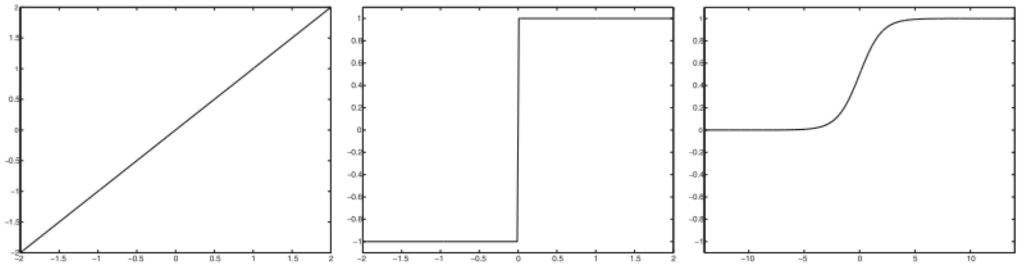
$$\Phi\left(v\right) = \frac{1}{1 + e^{-v}} \tag{2.19}$$

$$\Phi\left(v\right) = \frac{e^{2v} - 1}{e^{2v} + 1} \tag{2.20}$$

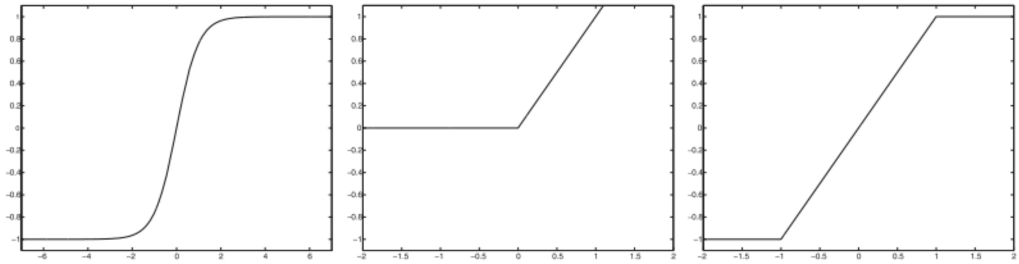
$$\Phi\left(v\right) = \begin{cases} -1 & \text{if } v < -1 \\ v & \text{if } -1 \leq v \leq 1 \\ 1 & \text{if } v > 1 \end{cases} \tag{2.21}$$

There are many activation functions, each of which transforms the output in a different way, modifying the network's behavior to better suit different applications. For example, Equations (2.18), (2.19), (2.20), and (2.21) are well suited for classification tasks, as their outputs are constrained to high or low values, indicating whether the input sample belongs to a class or not.

All the activation functions mentioned are illustrated in Figure 2.24.



(a) Identity (b) Sign (c) Sigmoid



(d) Tanh (e) ReLU (f) Hard Tanh

Figure 2.24: Various activation functions. C. C. Aggarwal, 2018 [58].

2.3.2.2 Loss Function and Optimization

The training process relies on minimizing the prediction error. The objective of the perceptron can be written in least-squares form with respect to all training instances in a dataset $\mathcal{D}$ containing feature-label pairs [58]:

$$\text{Minimize}_{\bar{\mathbf{W}}} L = \sum_{\left(\bar{\mathbf{X}}, y\right) \in \mathcal{D}} (y - \hat{y})^2 = \sum_{\left(\bar{\mathbf{X}}, y\right) \in \mathcal{D}} \left(y - \left\{\bar{\mathbf{W}} \cdot \bar{\mathbf{X}}\right\}\right)^2 . \tag{2.22}$$

This type of minimization objective is also referred to as a *loss function*, in this case based on the Sum of Squared Errors (SSE) [58]. Specifically, the loss function measures the difference between the reference and predicted labels, but it can be designed to evaluate any target characteristic available for training, thereby optimizing the model with respect to a specific behavior using one or more variables.

There are different training algorithms, usually referred to as *optimizers*, all of which work by feeding each input data instance $\bar{\mathbf{X}}$ into the network one by one (or in small batches) to generate a prediction $\hat{y}$. The weight vector $\bar{\mathbf{W}}$ is updated at each iteration based on the error value given by $E(\bar{\mathbf{X}}) = (y - \hat{y})$. This update is represented in Equation 2.23.

$$\bar{\mathbf{W}} \Leftarrow \bar{\mathbf{W}} + \alpha \left(y - \hat{y}\right) \bar{\mathbf{X}} \tag{2.23}$$

where: α = Learning rate.

Training algorithms repeatedly cycle through all the training examples in random order and iteratively adjust the weights until a certain condition is met (for example, a fixed number of iterations, a target error value, or minimal change in error between iterations). A single training data point may be used multiple times. Each such cycle is referred to as an *epoch* [58].

2.3.3 Multi-layer Neural Network

The perceptron (basic unit of a neural network) is a *linear model*, in which the equation $\bar{\mathbf{W}} \cdot \bar{\mathbf{X}} = 0$ defines a linear hyperplane. The values of $\bar{\mathbf{W}} \cdot \bar{\mathbf{X}}$ can be positive or negative depending on whether $\bar{\mathbf{X}}$ lies on one side or the other of the hyperplane. This type of model performs particularly well when the data is *linearly separable* [58].

Examples of linearly separable and non-linearly separable data are shown in Figure 2.25.

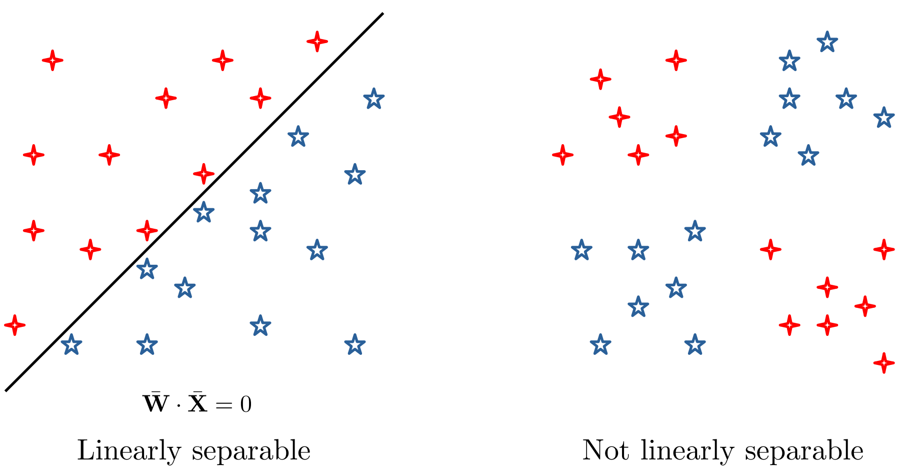
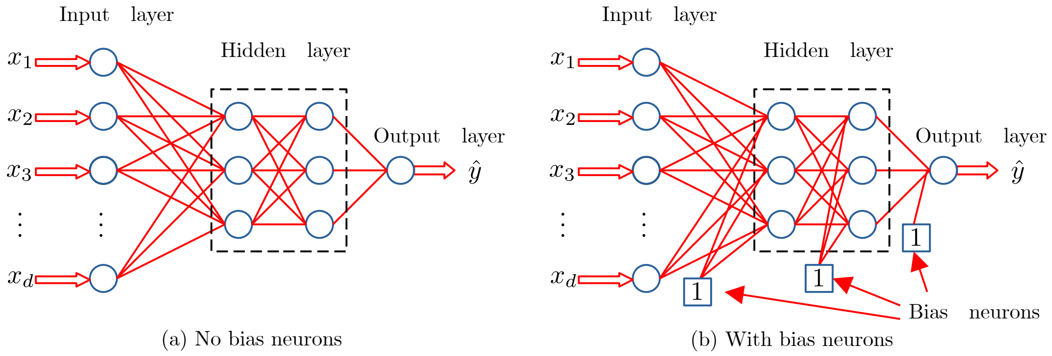


Figure 2.25: Examples of linearly separable and non-linearly separable data for two classes.

This limitation is analogous to fitting a linear function to a dataset. To address this, the combination of linear models (or multiple-layered interconnected perceptrons) can be interpreted as an extension from linear models to polynomial or more complex functions. By stacking layers and applying nonlinear activation functions, *multi-layer neural networks* can learn to approximate a wide range of nonlinear mappings. Each additional layer allows the network to compose abstract features from the raw input, enabling it to capture the complex structures required to solve non-linearly separable problems.

Multi-layer architectures contain more than one computation layer. The additional computation layers (between input and output) are referred to as *hidden layers* because the computations performed are not visible to the user. These networks are also known as *feed-forward* networks, as successive layers feed into one another in the forward direction from input to output. ”The default architecture of feed-forward networks assumes that all nodes in one layer are connected to those of the next layer. Therefore, the architecture of the neural networks is almost fully defined, once the number of layers and the number/type of nodes in each layer have been defined” [58]. Examples of multilayer networks with and without bias neurons are shown in Figure 2.26. The network shown contains three layers. Note that the input layer is often not counted, as it simply transmits the data and performs no computation.



(a) No bias neurons

(b) With bias neurons

Figure 2.26: Basic architecture of a feed-forward network with two hidden layers an a single output layer.

The hidden layers, also known as Fully Connected (FC) layers, apply a linear transformation between the input layer data $\mathbf{X}^{(l-1)}$ and the layer parameters, weights $\mathbf{W}^{(l)}$, and biases $b^{(l)}$:

$$\mathbf{X}^{(l)} = \mathbf{W}^{(l)} \cdot \mathbf{X}^{(l-1)} + b^{(l)} \tag{2.24}$$

The number of parameters can be calculated as the sum of all the connections between adjacent layers [68]:

$$n_{parameters} = \sum_{i=0}^{L-1} \left(n_{nodes}^{(l)} \cdot n_{nodes}^{(l+1)} + 1 \right) \tag{2.25}$$

The input data and the extracted features are limited to a vector representation, which leads to the loss of some spatial-contextual information relevant to remote sensing imagery [68].

2.3.4 Convolutional Neural Network

CNNs are biologically inspired models based on human vision. These networks have been widely used in computer vision for classification tasks [58]. One of the first basic architectures inspired by this biological concept was the *neocognitron*, which was later generalized into the *LeNet-5* architecture (Figure 2.27) [69].

There are two main types of layers in CNNs: *convolutional* and *subsampling* layers. In convolutional layers, a *convolution operation* is defined in which a filter is used to map the activations from one layer to the next. A convolution operation uses a filter (also known as a kernel) with the same depth as the input layer but with a smaller spatial extent. The dot product between all the weights in the filter and a spatial region of the same size in the input layer defines the value of the hidden state in the next layer (after applying an activation function). This operation is repeated across all possible positions in the input to compute the output of the next layer [58].

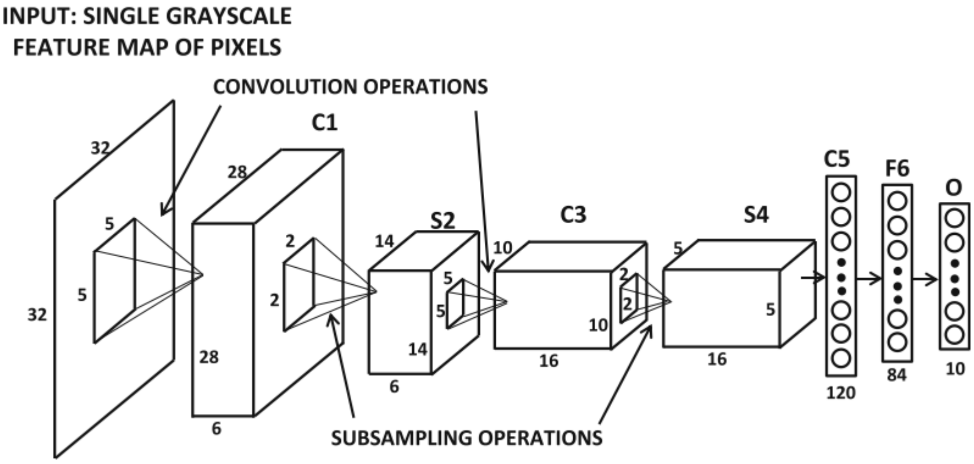


Figure 2.27: LeNet-5: One of the earliest CNNs. C. C. Aggarwal, 2018 [58].

2.3.4.1 Convolutional Layers

Spectral imagery classification relies on analyzing the statistical properties of $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{nI \times J \times n_{bands}}$, which can be considered a stationary source of spectral pixels, where data features are distributed across the image according to spatial positions. This suggests that the learned features can be generalized for predicting other pixels in different images with similar data characteristics [68].

In contrast to FC layers, Convolutional (CONV) layers offer greater versatility, as the size of the chunks or windows is defined by the receptive field of the layer, denoted as $k^{(l)} \times k^{(l)} \times q^{(l)}$, where $k^{(l)}$ is applied over the two spatial axes and $q^{(l)}$ over the spectral axis. This flexibility allows CONV layers to accept 1-D, 2-D, and 3-D inputs and to extract spatial, spectral, or spatial-spectral features.

The l-th CONV layer applies $K^{(l)}$ linear 3D kernels over the input layer $\mathbf{X}^{(l-1)}$, performing a dot product between its weights and biases, $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$, respectively, and small chunks of the input volume data. As a result, an output volume $\mathbf{X}^{(l)}$ composed of $K^{(l)}$ feature volumes is obtained [68].

$$\mathbf{X}^{(l)} = \left(\mathbf{W}^{(l)} * \mathbf{X}^{(l-1)} + b^{(l)}\right)_{K^{(l)} \times k^{(l)} \times k^{(l)} \times q^{(l)}} \tag{2.26}$$

The following equation indicates the general calculation of the feature at position (i, j, t) for the z-th feature map of the output volume, denoted as $x^{(l)z}i, j, t$.

$$x^{(l)_z}_{i,j,t} = \sum_{\hat{i}=0}^{k^{(l)}-1} \sum_{\hat{j}=0}^{k^{(l)}-1} \sum_{\hat{t}=0}^{q^{(l)}-1} \left(w^{(l)}_{\hat{i},\hat{j},\hat{t}} \cdot x^{(l-1)}_{\left(i \cdot s^{(l)}+\hat{i}\right),\left(j \cdot s^{(l)}+\hat{j}\right),\left(t \cdot s^{(l)}+\hat{t}\right)} \right) + b^{(l)} \tag{2.27}$$

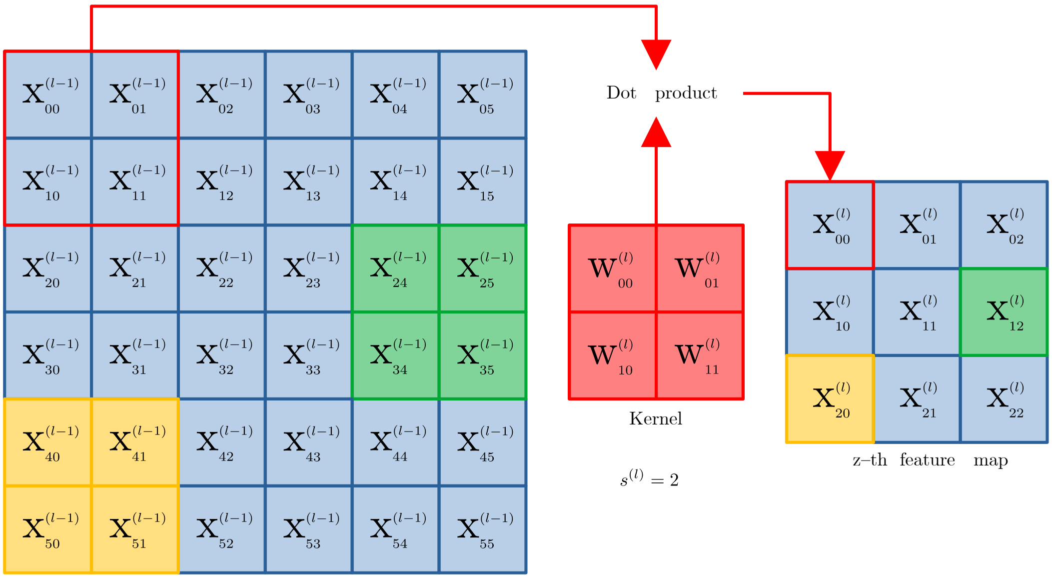


Figure 2.28: Graphical visualization of the CONV layer from a 2D point of view.

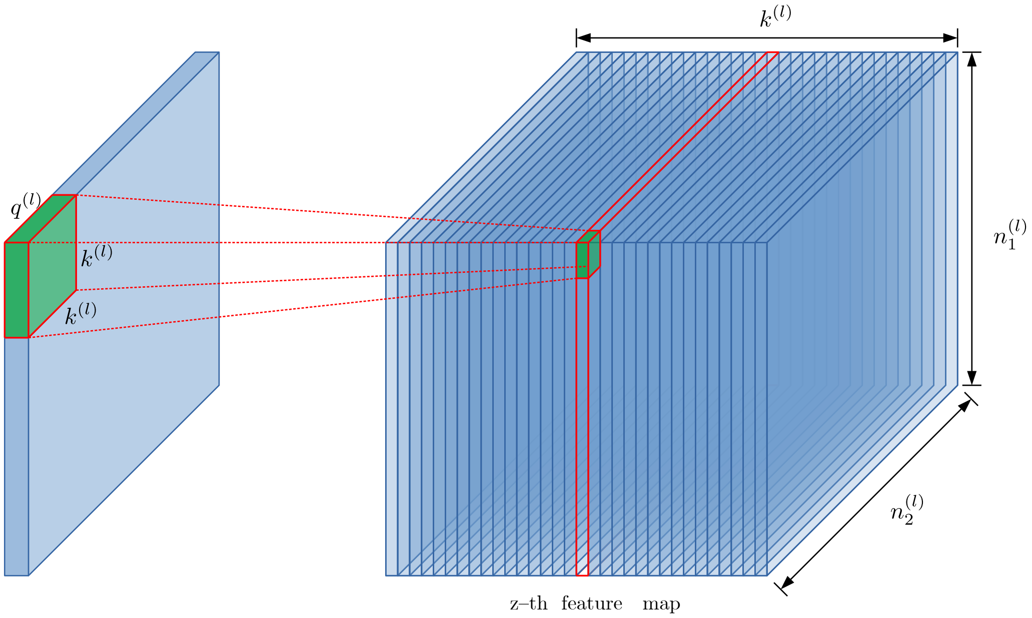


Figure 2.29: Graphical visualization of the CONV layer from a 3D point of view.

The architecture of a CNN used for spectral imagery classification is composed of two well-differentiated parts that can be interpreted as two separate networks:

- The Feature Extraction Network (FE-Net), composed of a hierarchical stack of feature extraction and detection stages that learn high-level representations of the inputs.
- The classifier, composed of a stack of FC layers that performs the final classification task by computing the membership of each input sample to a specific class [70].

These coupled networks are trained together as an end-to-end model to optimize all the weights in the CNN. Focusing on the classifier network, it performs the final classification by taking into account the information obtained by the FE-Net. This part is usually implemented with several stages composed of FC and ReLU layers, ending with an activation function applied to the last FC layer [68].

The output of the network can be interpreted as a confidence index indicating the likelihood that the input data (a pixel) belongs to a given class. The optimization function used for this task is the *cross-entropy*, defined in Equation 2.28, which measures the difference between $\mathbf{y}_i$ (the reference label of each pixel) and $\hat{\mathbf{y}}_i$ (the predicted values) [68].

$$\Phi_c = -\sum_i \hat{\mathbf{y}}_i - \log\left(\mathbf{y}_i\right) \tag{2.28}$$

2.3.4.2 Down-sampling Layers

Also known as Pooling (POOL) layers, they are inspired by the spatial processing of CONV layers and perform a non-linear sub-sampling strategy with three main objectives [68]:

- Reducing the spatial dimensions of the extracted feature maps by summarizing them into a reduced volume.
- Providing a degree of invariance to small transformations in the input data.
- Reducing computation time and complexity in terms of both data size/dimensionality and the number of network parameters [71].

The POOL layer implements a sample-based discretization process (Figure 2.30), where the dark cells may represent the selected values in a max-pooling operation. However, other pooling strategies exist, such as computing the average or the sum of the values within a region of pixels. In fact, the pooling layer can be interpreted as a kernel of size $k^{(l)} \times k^{(l)}$[68]. The most common operations are average pooling, sum pooling, and max pooling [72].

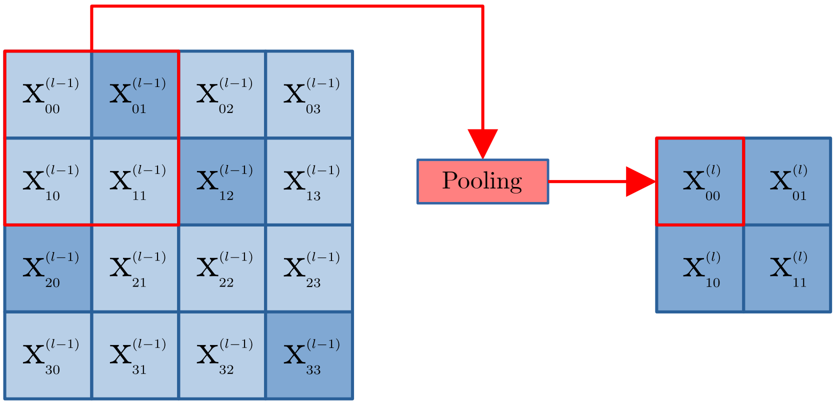


Figure 2.30: Graphical visualization of the POOL layer from a 2D point of view.

2.3.5 CNNs for Spectral Imagery Feature Analysis

Among many different architectures for pixel-based classification, three types of CNN models are addressed in this thesis for optical remote sensing imagery classification. These are categorized based on whether they perform spectral, spatial, or spectral–spatial feature extraction [68].

2.3.5.1 CNN1D for Spectral Feature Extraction

An backbone of a 1D - Convolutional Neural Network (CNN1D) architecture is shown in Figure 2.31, where the spectral pixels $\mathbf{x}i \in \mathbb{N}^{nbands}$ are the input data, where n_{bands} is the number of bands in the image, with or without compression. One-dimensional kernels of size $K^{(l)} \times q^{(l)}$ are applied at each CONV layer, resulting in an output $\mathbf{X}^{(l)}$ composed of $K^{(l)}$ feature vectors [68].

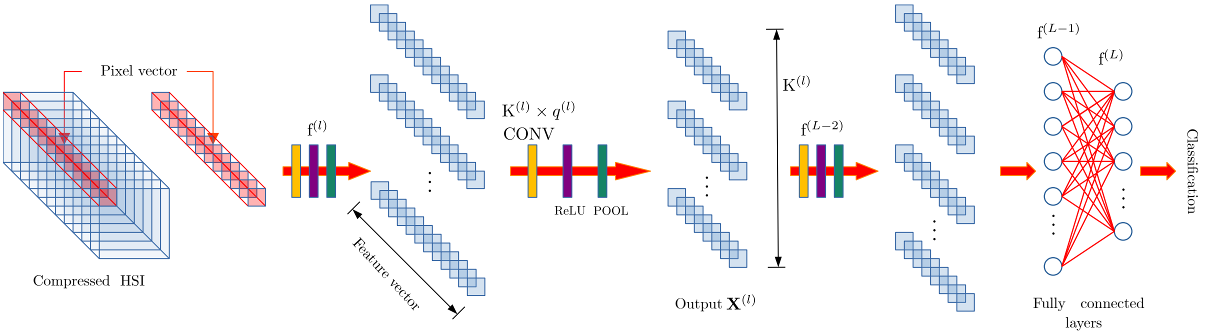


Figure 2.31: Architecture of spectral convolutional model employed by CNN1D.

2.3.5.2 CNN2D for Spatial Feature Extraction

These CNN models consider the spatial information obtained from spectral images. To process spatial information, each CONV layer applies $K^{(l)} \times k^{(l)} \times k^{(l)}$ kernels over the input data, resulting in $K^{(l)}$ feature maps (see a backbone of a 2D - Convolutional Neural Network (CNN2D) architecture in Figure 2.32). The spatial information is extracted by cropping spatial patches of $d \times d$ pixel-centered neighborhoods from the spectral image [68]. After obtaining the feature maps for each band, these are used as input to the FC layer, which performs classification based on the relationships between the spatial features extracted across bands.

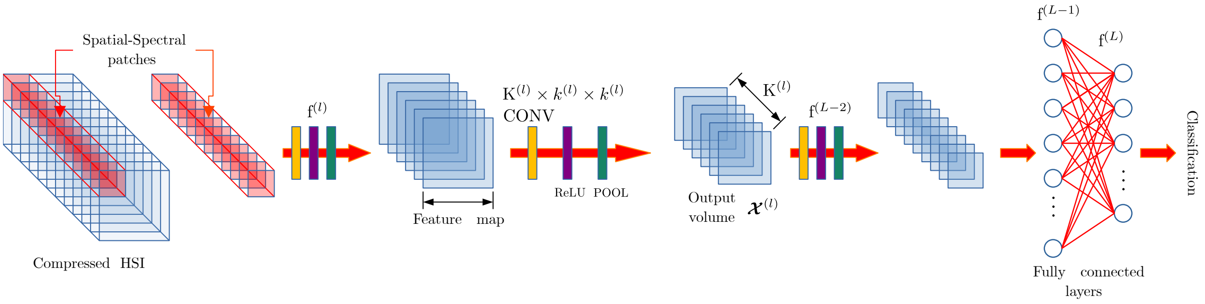


Figure 2.32: Architecture of spatial convolutional model employed by CNN2D.

2.3.5.3 CNN3D for Spectral-Spatial Feature Extraction

The 3D - Convolutional Neural Network (CNN3D) model (backbone shown in Figure 2.33) is naturally adapted to optical remote sensing imagery, where spatial and spectral correlations are typically high. It performs convolution using $K^{(l)}$ 3D filters, such that $K^{(l)} \times k^{(l)} \times k^{(l)} \times q^{(l)}$, which are used to extract 3D features. These filters simultaneously capture spatial information from neighboring pixels and spectral information from contiguous bands, producing as output $K^{(l)}$ feature volumes [68].

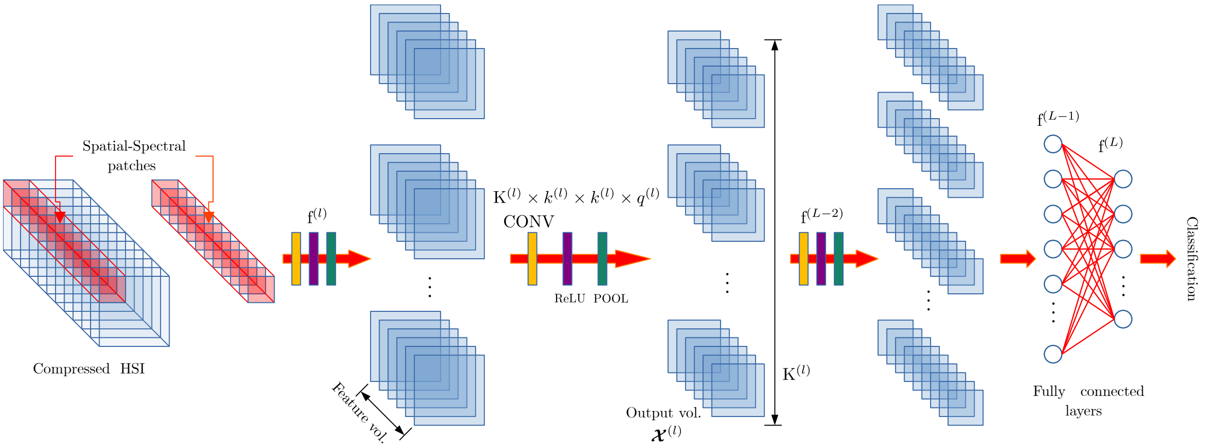


Figure 2.33: Architecture of spectral-spatial convolutional model employed by CNN3D.

The parameter definitions of the aforementioned architectures (such as kernel size, the number of CONV and POOL layers, and the window size for pixel patches) depend on various characteristics of the input image, including spatial resolution, number of bands, and the observed target. For example, agricultural crops tend to exhibit strong spatial correlation due to vegetation uniformity. In contrast, urban areas present the opposite case, where the captured targets are smaller and typically lack uniformity.

Therefore, the model parameters are usually selected empirically or, when sufficient computational resources are available, through an exhaustive search strategy. This approach aims to identify the parameters that enable the extraction of the most meaningful features by evaluating classification performance on a validation dataset.

2.4 Related Work

This section will revise the topics regarding the related work in three different context:

- Domain generalization for global-scale implementation (Subsection 2.4.1)
- An intercomparison exercise for revision of classifiers in the same application scenario (Subsection 2.4.2)
- State-of-the-art neural network-based classifiers (Subsection 2.4.3)

2.4.1 Domain Generalization

"Machine learning systems generally assume that the training and testing distributions are the same. To this end, a key requirement is to develop models that can generalize to unseen distributions". "Domain generalization deals with a challenging setting where one or several different but related domain(s) are given, and the goal is to learn a model that can generalize to an unseen test domain." [73].

Given training samples (x_i, y_i) for $i = 1, \dots, n$ and a loss function $L(h(x), y)$, where $h(x)$ is the hypothesis function. From a classical perspective view for model's fitting, there is a trade-off when minimizing the following risks [74]:

- Empirical Risk Minimization (ERM) involves obtaining a function from an hypothesis space that minimizes the average loss over the training data, such that:

$$R_{\text{emp}}(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(x_i), y_i) \tag{2.29}$$

Therefore, the hypothesis h^* is obtained by:

$$h^* = \arg\min_{h \in \mathbb{H}} R_{\text{emp}}(h) \tag{2.30}$$

This approach assumes that by minimizing the error, the hypothesis function will generalize well for the unseen data, but this depend on the data distribution of the testing dataset. In single-image analysis or single-site analysis remote sensing classification implementation, ERM is a well-suitable approach, given that the variation of samples in the domain is small, even allowing to perform manual discarding of samples (e.g. discarding cloudy scenes), making overfitting not as harmful as when the domain is difficult to represent, e.g. for global-scale applications.

- Structural Risk Minimization (SRM) in contrast, deliberately controls the model training to prevent overfitting, aiming to achieve high generalization. In other words, underfitting the models to achieve robustness. This is represented by a regularization term that penalizes complexity (or low training error) and weighted by a parameter which controls the trade-off between empirical and structural risk. , defined as follows [74]:

$$R_{\text{struct}}(h) = R_{\text{emp}}(h) + \lambda \cdot \Omega(h) \quad h^* = \arg\min_{h \in \mathbb{H}} R_{\text{struct}}(h) \tag{2.31}$$

where: $\Omega(h)$ = Regularization term which measures hypothesis overfitting
λ = Regularization parameter

Specially useful when training and test sets differ in distribution (domain shift).

In the context of optical remote sensing missions with global coverage, a classifier must be able to generalize across diverse domains. This thesis addresses this challenge by training deep learning models on data from multiple geographic locations and under varying atmospheric conditions. Early stopping criteria were determined through a grid search over the training hyperparameters. While this represents a fundamental approach, it typically requires several iterations and careful interpretation of inference outcomes.

A natural next step is to incorporate a regularization term into the training loss function that promotes the retrieval of physically meaningful parameters, thereby improving the stability and generalizability of the training process. However, this introduces new challenges—such as the need to fit a classifier per class or for groups of physically related classes—since physical parameters are often specific to distinct observable phenomena in remote sensing imagery. These considerations are further discussed in Section 4.3.

Recent works in remote sensing have explored large-scale classification using self-supervised or weakly supervised approaches, often leveraging unlabeled or physically derived data. Notably, Tuia et al. [75] provide a perspective on physically informed learning as a key direction for advancing Earth observation. In related work [76, 77, 78, 79, 80], the Copernicus land cover product CORINE [81] has been used in conjunction with BigEarthNet [82] to train classification models while explicitly accounting for label noise. Our work follows a similar trend by combining physically based label generation with learning-based refinement, applied to globally distributed Sentinel-2 data. One of the few works that incorporate information from spectral indices or derived masks for training is the work presented in [83] on water applications to delineate water bodies from widely used water spectral indices.

2.4.2 Inter comparison Exercises

For the review of the current state of classification algorithms in optical remote sensing, specifically for Sentinel-2, the latest Cloud Mask Intercomparison eXercise (CMIX) [84] is considered the main reference. This is an international collaborative effort initiated within the framework of the Committee on Earth Observation Satellites (CEOS) Working Group on Calibration and Validation (WGCV), with participation from public and private entities. Although this exercise focuses on cloud masking, the classification algorithms evaluated in CMIX classify pixels into several classes, such as clear sky, water, snow, cloud shadows, thin clouds, and more. Table 2.5 provides a brief summary of the participating processors from the latest edition of CMIX.

It is important to note that some of the algorithms listed in this table have already been implemented in processing chains of remote sensing missions, thus enabling global-scale implementation. In these kinds of exercise, the reference datasets were independently and manually labeled and are not considered in the fitting process of any of the algorithms, unlike most state-of-the-art classifiers, which typically split a single dataset into training and test sets. The unique-labeling-source approach meets the needs of the dataset's specific problem by using labels based on strict specifications. However, this method is not fully applicable to generic classification products of remote sensing missions with global coverage, given the high variability of subclasses.

There are three major areas on which the classification techniques are based: spectral tests (or spectral indices thresholding), machine learning, and multi-temporal analysis. Those algorithms based on spectral tests, which run test on empirically defined thresholds tested over computed indices of different bands combination of the spectral image. Each of the indices is designed to highlight one of the physical properties of the materials that interact with the light in the Earth's surface or the atmosphere [29]. Among those in Table 2.5, ATCOR stands out as one of the most established atmospheric correction algorithms, forming the basis for PACO, which is the atmospheric correction processor used in this thesis.

Machine learning approaches fit different types of artificial neural network models with high-quality data. The main challenge for training is to obtain high-quality classification maps on a global scale with sufficient representation of all types of Earth's surface materials under different atmospheric conditions. In addition, this process must be performed for every sensor that is intended to be supported. Alternatively, the support to multispectral and hyperspectral sensors arises a need to transform the data into an equivalent space to generalize the model for a variety of sensors, usually resulting in a loss of spectral information. This is the approach adopted in this thesis, and the methodology will be suited in the state-of-the-art in the following Subsection 2.4.4.

For multi-temporal analysis, repetitive observations with a short time difference between captures at similar viewing angles are needed to compare changes in reflectance with the background, obtaining promising results. However, these approaches are limited

Processor	**Organization**	**Methodology**	**Application**
ATCOR [29, 49]	DLR	Spectral tests	Multi-/ hyper-spectral
CD-FCNN [85, 86]	University of Valencia	Machine learning	Multi-spectral
Fmask [87, 88, 89]	USGS	Spectral tests	Multi-spectral
FORCE [90, 91, 92]	Humboldt-Universit¨at zu Berlin / Trier University	Spectral tests	Multi-spectral
Idepix [93]	Brockmann Consult	Spectral tests	Multi-spectral
S2cloudless [94]	Sinergise	Machine learning	Multi-spectral
InterSSIM [95]	Sinergise	Machine learning and spatio-temporal context	Multi-spectral
LaSRC [7, 96, 97]	NASA / University of Maryland	Spectral tests	Multi-spectral
MAJA [98, 99]	CNES / CESBIO	Multi-temporal data and spectral tests	Multi-spectral
Sen2Cor [100, 101]	ESA / Telespazio France	Spectral test and auxiliary data	Multi-spectral

Table 2.5: Participant masking algorithms in the last edition of CMIX. Table extracted from [84].

in their implementation due to the need for a location's temporal series.

It is not possible to declare a single best classification algorithm, because the results depend on the dataset being tested. The main issue is that there is no consensus on the quantitative definition of clouds. As a result, every dataset has a bias that favors certain algorithms. For example, the Atmospheric Correction Algorithm (ATCOR), CD-FCNN, InterSSSIM, LaSRC, and Sen2Cor have been shown to be cloud conservative, meaning these algorithms detect fewer clouds over clear regions, at the expense of missing some clouds. Conversely, Fmask, FORCE, Idepix, MAJA, and S2cloudless provide a balance between commission and omission errors, but at the cost of also classifying out valid, non-cloudy observations.

The results reported in [84] show that machine learning-based masking algorithms do not perform significantly better than spectral test-based algorithms, as their predictions depend on the data quality used for training, and these datasets are not available for all sensors. Since the spectral tests of currently used masking algorithms typically do not consider all available spectral bands, DL-based classification algorithms could potentially identify hidden patterns within spectral signatures, but only if the training pixels are correctly labeled.

Future CMIX activities plan to test the processors with hyperspectral datasets, encouraging processors to support hyperspectral products.

2.4.3 Classifiers State-of-the-art based on Neural Networks

Recently published methods for pixel-level classification, which exploit the growing area of DL models [102], have recently been applied to remote sensing applications, usually based on CNNs [68, 103], Generative Adversarial Networks (GANs) [104, 105, 106], or Transformers [107, 108, 109] among other approaches. In addition, the use of attention strategies [110, 111] has shown advantages in identifying fine features to improve classification performance on single-image datasets. All these works show promising results, and it is expected that emerging large models will make a huge impact on classification performance, as demonstrated in other study areas in recent years. However, the lack of high-quality harmonized labeled data limits researchers and developers to using datasets with the same labeling source for both training and testing sets. This approach on training with a splitted dataset intro training and testing demonstrates the model's capability of differentiating between the specified classes in the dataset (the requirements of the specific application are described in the labels), but it is limited by the sensitivity of the labeling source, and therefore, not demonstrating generalization for other applications. Thus, this restriction limits the interpretability of the results for global-scale remote sensing missions.

DL models based on CNNs for pixel-level classification are able to retrieve more spatial and spectral features compared to classic masking algorithms based on physics models. This is because they analyze high data volumes, considering all the spectral signatures and the information from neighboring pixels [112]. However, they highly dependent on the data and reference labels for training, with the added challenge of interpreting the retrieved features. Although models like CNN3D [113] can simultaneously extract spatial and spectral features, they also increase computational complexity.

2.4.4 Methodological Scope

In this work, three lightweight architectures are compared: CNN1D [114], CNN2D [112, 115], and CNN3D [113], which respectively represent purely spectral, spatial–spectral, and fully joint spectral–spatial feature extractors. These models have demonstrated solid performance compared to traditional machine learning approaches [68]. Encoder–decoder architectures such as U-Net [116] are typically designed for dense ground-truth masks; however, the supervision strategy in this thesis explicitly excludes ambiguous pixels, which do not contribute to the loss function or its backpropagation [117], making U-Net less suitable for the chosen setting. The selected CNN1D, CNN2D, and CNN3D architectures span a representative spectrum of modeling capacities, aligned with the taxonomy proposed by Paoletti et al. [68].

Advanced methods based on GANs, transformers, or attention mechanisms are not included in this study. Their evaluation is reserved for future work, as the primary goal here is to validate the general concept introduced in [118], specifically to improve training datasets through conservative, physics-based pixel selection strategies, and to provide a detailed methodology and robust evaluation of the results. The core motivation for adopting deep learning in this work stems from the limitations of the current PACO masking algorithm, which relies on simple spectral thresholding and does not exploit complex spatial–spectral patterns that CNNs are well-equipped to model.

For classification evaluation, widely used metrics such as Overall Accuracy (OA) and F-1 score (F1) have been shown to be unreliable for imbalanced datasets [119, 120], as they become biased when class distributions are uneven. Therefore, more robust alternatives are required, such as the nMCC, which is detailed in Subsection 3.8.1. nMCC corresponds to the R_k statistic for the multi-class case and to the Phi coefficient in the binary case. In addition, confusion matrices and class-wise precision (Precision (P)) and recall (Recall (R)) are included to support a more comprehensive interpretation of performance across individual classes.

Chapter 3

Methodology

In this chapter, the methodology developed for training and evaluating CNNs in the context of operational masking tasks is presented. The approach leverages existing classification products derived from operational algorithms as training references, while validation is performed using independent data sources, including products from external sources. This strategy enhances the generalization capability of the models and ensures a more robust and less biased evaluation framework. Therefore, this work addresses the pixel-level classification task with a supervised deep learning model fitted with the current masks of PACO currently derived from spectral indices. To improve classification performance, the model must extract more significant spatial and spectral features, as well as the correlations between them, than those used by the current labeling source algorithm. This enables it to redefine the classes in the new classification space of the fitted model, differentiate between the classes, and thereby improve performance. In this section, the main characteristics of the implemented methodology are presented and discussed, covering the notation used, the problem statement together with the big picture of the implemented workflow (shown in Fig. 3.1). At the end, the guidelines followed throughout the process are described, including data preprocessing, pixel selection, retrieval of higher-quality pixels for training, and finally, the model architecture and fitting details, along with its evaluation.

Using the notation from [28] and the representation of spectral cubes with tensors, where a is a scalar, $\mathbf{a}$ is a vector, $\mathbf{A}$ is a matrix, and $\boldsymbol{\mathcal{A}}$ is a 3^{rd}- or higher-order tensor, a spectral image of an observed scene $\boldsymbol{\mathcal{X}}$ is represented as a 3^{rd}-order tensor:

$$\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}, \tag{3.1}$$

where I (rows) and J (columns) are the spatial dimensions of the image, and K is the number of spectral bands. Then, the element $x_{i,j,k}$ is the pixel value at position (i, j) in band k, such that:

$$\{\forall i, j, k \in \mathbb{Z} \mid (1 \leq i \leq I) \wedge (1 \leq j \leq J) \wedge (1 \leq k \leq K)\}. \tag{3.2}$$

and the vector containing the spectral signature of a pixel is represented by $x_{i,j,:}$.

Each ordered pair (i, j) represents the coordinates of a specific location on the Earth's surface in the observed scene.

A classification algorithm is defined as a function:

$$f_{\text{multi-class}} : \boldsymbol{\mathcal{X}} \rightarrow \hat{\mathbf{Y}}, \hat{\mathbf{Y}} \in \mathbb{Z}^{I \times J}, \tag{3.3}$$

which maps from the spectral image $\boldsymbol{\mathcal{X}}$ to the predicted classification map $\hat{\mathbf{Y}}$, matching the spatial resolution. This process is also known as pixel-based multi-class classification, where each element $\hat{y}_{i,j}$ of $\hat{\mathbf{Y}}$, represents a positive integer, so-called label, indicating the class to which it belongs.

Optionally, the masks could also be provided in a binary cube for multi-label classification (masks cube), such that:

$$f_{\text{multi-label}} : \boldsymbol{\mathcal{X}} \rightarrow \hat{\boldsymbol{\mathcal{Y}}}, \hat{\boldsymbol{\mathcal{Y}}} \in \{0, 1\}^{I \times J \times C}, \tag{3.4}$$

where C is the number of classes, allowing a pixel to belong to more than one class. Furthermore, $\hat{\boldsymbol{\mathcal{Y}}}_{\text{ci}}$ represents the confidence index (denoted in the subscript as "ci") of each pixel belonging to a class, such that:

$$f_{\text{confidence-index}} : \boldsymbol{\mathcal{X}} \rightarrow \hat{\boldsymbol{\mathcal{Y}}}_{\text{ci}}, \hat{\boldsymbol{\mathcal{Y}}}_{\text{ci}} \in \mathbb{R}^{I \times J \times C}, \tag{3.5}$$

where each of the elements $\hat{y}^{\text{ci}}_{i,j,c}$ follows the axioms of probability, such that:

$$\left(0 \leq \hat{y}^{\text{ci}}_{i,j,c} \leq 1\right), \tag{3.6}$$

$$\sum_{c=1}^{C} \hat{y}^{\text{ci}}_{i,j,c} = 1. \tag{3.7}$$

The term confidence index is used instead of probability because the output of the last layer of a neural network is generated by activation functions, not by PDFs, although the general interpretation is similar.

3.1 Problem Statement

The following section provides a general description of the inputs, outputs, and processes of the proposed methodology in this thesis. Each product and steps are summarized in Figure 3.1, which presents an overview of the workflow.

The main objective of this thesis is to obtain a classifier based on a Deep Learning (DL) model such that:

$$f_{\text{DL}} : \boldsymbol{\mathcal{X}} \rightarrow \hat{\boldsymbol{\mathcal{Y}}}_{\text{ci}}, \tag{3.8}$$

where $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times F}$, is a cube containing the spectral data, DEM, and illumination, normalized and stacked into one cube. Here, I and J correspond to the rows and columns of the image, respectively, and F to the number of features. These dimensions correspond to the input tensor $\boldsymbol{\mathcal{X}}$, illustrated as the purple cube in Figure 3.1.

Given that the classification is performed pixel-wise, the input size of the model is a small patch $\boldsymbol{\mathcal{X}}_{\text{patch}} \in \mathbb{R}^{W \times W \times F}$, where the window size W is odd to ensure that the pixel of interest is located at the center of the patch cube and surrounded by neighboring pixels, providing the spatial context for feature extraction. This is particularly helpful for high-spatial-resolution sensors, where the spatial correlation is high.

The output is a cube $\hat{\boldsymbol{\mathcal{Y}}}_{\text{ci}} \in \mathbb{R}^{I \times J \times C}$ containing the predicted confidence index, where C is the number of target classes. For the case of a multi-class classifier, $f_{\text{DL}} : \boldsymbol{\mathcal{X}} \rightarrow \hat{\mathbf{Y}}$, where $\hat{\mathbf{Y}} \in \mathbb{Z}^{I \times J}$ is a classification map, the highest confidence index from $\hat{y}^{\text{ci}}_{i,j,:}$ is selected as the most likely class, assigning a label represented as an integer in $\hat{\mathbf{Y}}$. These outputs are illustrated as blue matrices at the output stage of the DL model in Figure 3.1.

To fit the model f_{DL}, the starting point is the collection of an extensive database of scenes from a particular remote sensing monitoring mission to train the model. These scenes are pre-processed by the mission's atmospheric correction processor (PACO in this study), to obtain the TOA reflectance, DEM, and illumination products mentioned earlier. In addition, the masks cube $\boldsymbol{\mathcal{M}} \in 0, 1^{I \times J \times K}$ is obtained, where K is the number of classes from the source. The information from the multi-label product is then leveraged by filtering the pixels and keeping those with low classification uncertainty. Furthermore, the binary masks cube is mapped to the target set of classes, which contains C classes, usually $C \leq K$. The resulting cube is defined as $\boldsymbol{\mathcal{M}}_t \in 0, 1^{I \times J \times C}$, and it will be used by the optimizer to fit the model through supervised learning (see red-dashed box in Figure 3.1).

As a last step, for the testing phase, a representative database of scenes is required, where the reference labels are obtained from a reliable source using additional information, instruments, or expert-driven criteria to increase their sensitivity (see orange-dashed box in Figure 3.1). For example, each scene can be manually labeled by an independent human expert using diverse additional information sources to demonstrate the robustness and generalization of the trained model. In addition, measurements from accurate instruments can be used to compare the obtained products with the measured reference.

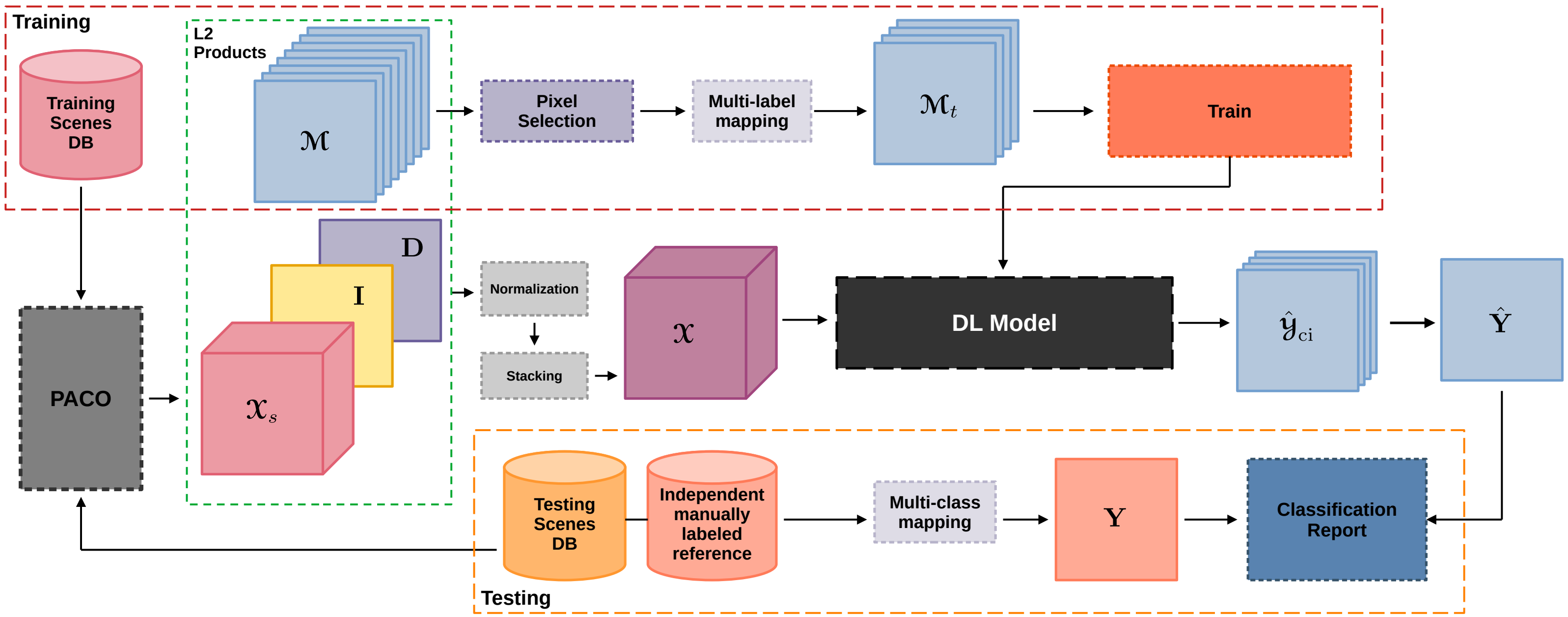


Figure 3.1: Workflow of the implemented deep learning-based approach for classification. For the training stage (upper section), training scenes are pre-processed using PACO to generate TOA reflectances ($\boldsymbol{\mathcal{X}}_s$), DEM ($\mathbf{D}$), and illumination ($\mathbf{I}$), which are normalized and stacked into $\boldsymbol{\mathcal{X}}$. The masks cube ($\boldsymbol{\mathcal{M}}$) is filtered by selecting pixels to create high-quality labels ($\boldsymbol{\mathcal{M}}_t$), used to train the DL model. For evaluation (lower section), independent testing datasets with manually labeled references ($\mathbf{Y}$) are compared against predictions ($\hat{\mathbf{Y}}$), generating a classification report with performance metrics.

3.2 Input L2 Products and Preprocessing

This Section describes the input data obtained from PACO and its normalization preprocessing of the DL model, together with the labels used for the training (see green-dashed box in Figure 3.1). As discussed in subsection 2.3.1.3, normalization ensures that the value range is consistent across all features. However, for the proposed methodology designed for continuous training, features were normalized by linearly re-projecting them into the range $[0, 1]$ as using Min-Max normalization. This was achieved by dividing each input dataset by its maximum expected theoretical value of the corresponding physical parameter.

For the spectral data, TOA reflectance values greater than 1 are not considered outliers, as they commonly occur when light is specularly reflected from different surfaces to the same captured pixel [121]. Thus, normalization is performed with a maximum reflectance of 1.5 as follows:

$$\mathcal{X}'_s \in \{x|0 \leq x \leq 1\}^{\text{Rows}\times\text{Columns}\times\text{Bands}} = \frac{\mathcal{X}_s}{1.5}. \tag{3.9}$$

The DEM was normalized using the highest point on Earth, Mount Everest, at 8850 m. Negative elevations were corrected to zero (e.g., in the Netherlands):

$$\mathbf{D}' \in \{d|0 \leq d \leq 1\}^{\text{Rows}\times\text{Columns}} = \frac{\mathbf{D}}{8850}. \tag{3.10}$$

Lastly, the illumination map, generated by PACO within the range $[0, 100]$, was normalized using its maximum expected value:

$$\mathbf{I}' \in \{i|0 \leq i \leq 1\}^{\text{Rows}\times\text{Columns}} = \frac{\mathbf{I}}{100}. \tag{3.11}$$

This fixed-value normalization approach enables the continuous addition of new scenes to the training database without requiring re-normalization based on updated maximum or minimum values.

The final normalized datasets are then stacked into a single data cube, such that:

$$stack(\mathcal{X}'_s, \mathbf{D}', \mathbf{I}') = \mathcal{X} \in \{x|0 \leq x \leq 1\}^{\text{Rows}\times\text{Columns}\times(\text{Bands}+2)}, \tag{3.12}$$

where elevation and illumination are included as additional features of the spectral bands (Bands).

In addition, the multi-label masks used to fit the DL model correspond to the L2 products also generated by PACO, and are defined as:

$$\mathcal{M} \in \{0, 1\}^{\text{Rows}\times\text{Columns}\times\text{Classes}} \tag{3.13}$$

where each frontal slice represents a binary mask for a specific class. These masks are subsequently used to select high-quality training pixels, as further detailed in Section 3.5.

3.3 Multi/Hyperspectral Case Studies

The classification methodology was performed in two different datasets: multi- and hyperspectral data, described in Subsections 3.3.1 and 3.3.2, respectively). The same study principle is applied to both types of data, but the training and validation steps differ depending on the available validation datasets. In both the multispectral and hyperspectral cases, a DL model was trained using PACO's products as input, processed at the early stages of atmospheric correction.

First, given the availability of data for testing, a multispectral case was addressed using data from the Sentinel-2 mission. Regarding compatibility with other sensors, although other missions processed with PACO exist, independent validation labels remain limited or are not directly compatible. Furthermore, other sources of information, such as in-situ measurements, are difficult to obtain on a global scale. Therefore, an alternative approach was adopted. The methodology was tested for the hyperspectral case using data from the EnMAP mission, but given the lack of standardized test datasets for EnMAP, a strategy was proposed to compare the estimated cloud fraction with observations from the Sentinel-5P mission. The main objective is to retrieve reference data from missions specifically designed to sense a particular characteristic of the atmosphere or Earth's surface and use it to validate the masking products of other missions, particularly those in optical remote sensing. To achieve this, it is necessary to implement an interface that transforms the masking products into physical properties that match the format of the mission products used as a reference. Both case studies are described in detail below.

3.3.1 Multispectral Case for Sentinel-2 Imagery

In Figure 3.2, the Unified Modeling Language (UML) activity diagram for the multispectral case is presented.

As mentioned before, the models were trained for this case using data from the Sentinel-2 mission (described in Subsection 2.2.1). The test datasets belong to other validation activities from the Sentinel-2 Optical Mission Performance Cluster (OPT-MPC) [122] and DLR's internal validation of the Fmask, ATCOR, and Sen2Cor processors [123]. These reference labels (usually referred to as the ground truth) are composed of the tuple $(\mathfrak{X}, \mathbf{Y})$ where $\mathfrak{X}$ is the spectral data and $\mathbf{Y}$ its associated reference labels, both with the same spatial dimensions.

The classifier's evaluation is accomplished by generating a classification report, which includes the confusion matrix and classification metrics that are robust to class imbalance.

This is considered the standard case, illustrated in Figure 3.1, given that most testing is conducted this way, following the example and guidelines proposed in the last CMIX [84] exercise, where testing is performed with independently labeled data.

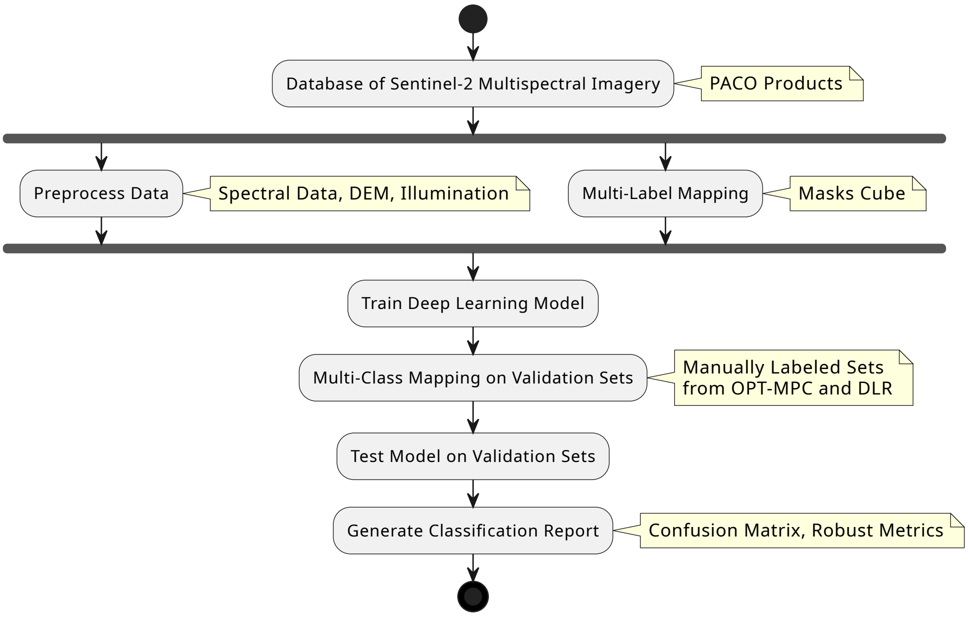


Figure 3.2: UML activity diagram of the multispectral classification case using Sentinel-2 and PACO products. Independent validation is performed with manually labeled datasets from OPT-MPC and DLR, allowing independent assessment using robust metrics.

3.3.2 Hyperspectral Case for EnMAP Imagery

In Figure 3.3, the UML activity diagram for the hyperspectral case is presented.

The hyperspectral case focuses on cloud classification, fitting a DL model with hyperspectral data from the EnMAP mission (described in Subsection 2.2.2) using exactly the same methodology as the multispectral case mentioned in subsection 3.3.1. The difference relies on the testing methodology, given that at the time of this thesis's development, no testing datasets were available for this hyperspectral mission.

As mentioned before, most testing exercises for masking algorithms or classifiers rely on hand-made annotations, but these require significant qualified manpower, given that the validation of each sensor requires the creation of a new dataset, such as the multispectral case described above. The lack of testing data for this case motivated the development of a testing dataset for this study, but this is complicated by the issue described in [96], which highlights misinterpretation of class definitions, often causing overlaps in annotations between classes.

Therefore, for the hyperspectral case, this study proposes a testing methodology that uses more sensitive and global sources of information, employing physical properties to validate classification products.

In this testing exercise, the cloud fraction is calculated from the cloud classification obtained from the models and compared with the reference retrieved from the TROPOMI

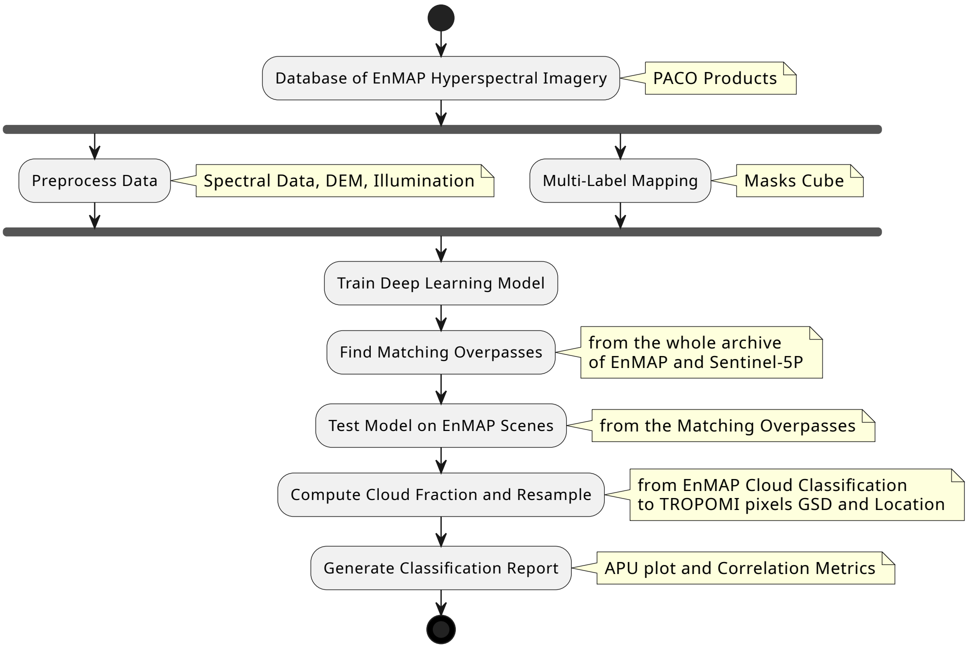


Figure 3.3: UML activity diagram of the hyperspectral classification case using EnMAP imagery and PACO products. Validation is performed by comparing cloud fraction from model predictions with Sentinel-5P observations, using matching overpasses and robust correlation metrics.

instrument onboard Sentinel-5P (described in Subsection 2.2.6.1) [56, 124], taking into account the availability of samples across different ranges of cloud fraction.

This methodology involves finding matching overpass locations with short time differences between the two missions, ideally on the order of seconds, but limited in practice to a maximum of three minutes due to the scarcity of overlapping acquisitions (see Section 3.4.4 for details). In Figure 3.4, an artistic visualization of a matching overpass location between EnMAP and Sentinel-5P is shown.

Thus, the EnMAP cloud mask (with a 30 m × 30 m resolution) is resampled to the spatial resolution of TROPOMI products (5.5 km × 3.5 km) by computing the CF of the EnMAP cloud mask projected into a TROPOMI pixel, and then it is compared with the CF product of TROPOMI using robust metrics and visualizations to analyze the differences. In Figure 3.5, a comparison between an EnMAP scene and the corresponding TROPOMI pixels is shown.

The details involved in calculating the CF from EnMAP scenes, including the resampling process via the projection of TROPOMI pixels, will be discussed further in subsection 3.4.4. This section will also address the challenges associated with minimizing uncertainty when comparing matching overpasses.

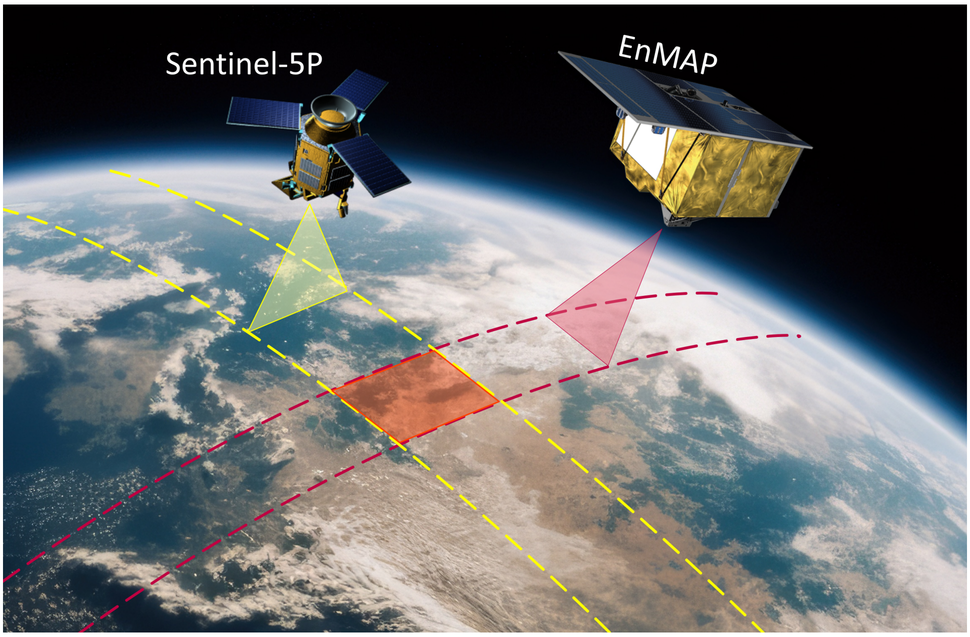


Figure 3.4: Illustration of a matching overpass between EnMAP and Sentinel-5P (not to scale). For cloud fraction comparison between the products of the TROPOMI instrument onboard Sentinel-5P and the predicted cloud mask from EnMAP, only matching overpasses with a small time difference should be considered (on the order of seconds), depending on wind speed and cloud altitude.

3.4 Datasets Description

DL approaches are designed to handle large data volumes for training, which enables the global-scale application of classification algorithms. Ideally, the training dataset should be as complete as possible to ensure representation of all potential atmospheric conditions across Earth's surfaces. However, the fact of not being diverse enough, or incorporating the entire archive of an optical remote sensing mission poses significant challenges and is computationally intensive. Therefore, a limited Database (DB) of representative L1 scenes has been created, with the potential for continuous expansion to support ongoing learning.

The datasets utilized in this thesis, both for training and testing, were obtained from the missions described in Subsections 2.2.1 and 2.2.2. However, the approach is not restricted to these missions, as it can be adapted to different spectral bands and spatial dimensions due to its pixel-level classification approach.

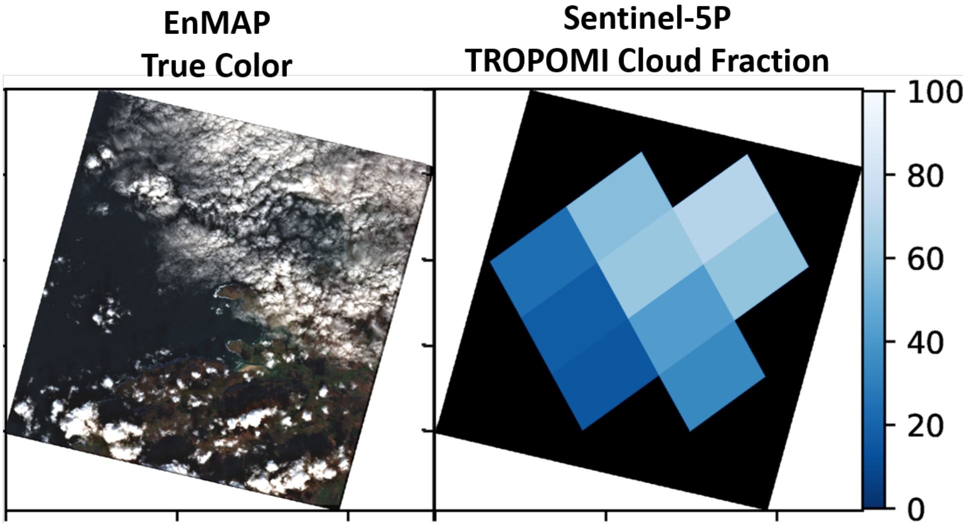


Figure 3.5: EnMAP true-color image (GSD: 30 m × 30 m) and TROPOMI CF product (GSD: 5.5 km × 3.5 km) comparison. EnMAP image: DT0000063937_004, captured on 2024-03-05 at 12:18:35 UTC.

3.4.1 Training Dataset for the Multispectral Case

The training dataset used in this case consists of 71 scenes selected to match the locations of the Aerosol Robotic Network (AERONET) sites [125], comprising $\approx 2,123$ million pixels distributed across the globe in a wide variety of biomes and atmospheric conditions. Additionally, a number of images were selected from the same geographic locations as the testing datasets; however, these correspond to different acquisition dates. Table 3.1 shows the pixel count by class for the training dataset.

Table 3.1: Training dataset class distribution and pixel count.

Color	Target Class	Pixel Count
🟫	Clear	69.92 %
▨	Atmosphere	11.41 %
⬛	Shadows	2.50 %
🟦	Water	12.81 %
⬜	Snow	3.37 %
	Total	2,122,676,571

As explained in Subsection 2.2.4, the labels for training are generated by PACO's masking, and these consist of 21 non-mutually exclusive masks, each corresponding to a pre-classification class. Therefore, a single pixel could belong to one or more masks, which in this work are also referred as a multi-label masks cube. These masks are represented in a cube $\mathcal{M} \in \{0,1\}^{5490\times5490\times21}$. A mapping is required to match the set of classes from the testing datasets. Consequently, five common classes are defined:

- *Clear* - Clear sky soil pixels
- *Atmosphere* - Semi-transparent clouds, clouds and cirrus
- *Shadows* - Shadows from clouds or terrain elevations
- *Water* - Seas, lakes, lagoons, rivers or any water body
- *Snow* - Permanent or temporal snow or ice

In addition, each mapping can incorporate the bias introduced by the subjective definition of each class (e.g., how transparent a cloud must be to be considered a semi-transparent cloud). Therefore, each mapping must be manually defined according to the specific definition of each class. Ideally, the classes should be defined in terms of physical metrics to harmonize classification products across different studies and applications. However, given the available testing datasets, a mapping has been defined for these specific cases. The first and second columns in Table 3.2 show the mapping from PACO pre-classification classes to the corresponding target classes addressed in this work, while the rest of the columns describe the mapping for the testing datasets, which will be explained in more detail in subsection 3.4.2. More details about the decisions made to define the mapping will be also reviewed in section 3.4.2. Further details on PACO masking and pre-classification can be found in [12].

The input dataset for the proposed classification algorithm is composed of three different sources pre-processed by PACO in the early stages of the atmospheric correction process and resampled to a resolution of 20 meters: the spectral information from the Sentinel-2 mission, denoted as $\mathcal{X}_s \in \mathbb{R}^{5490\times5490\times13}$; the DEM from the Copernicus Mission [53], $\mathbf{D} \in \mathbb{Z}^{5490\times5490}$ containing the altitude in MASL; and the corresponding illumination map $\mathbf{I} \in \mathbb{R}^{5490\times5490}$, produced using the geometrical relation between the sun light beam and the ground terrain surface.

	Training	**Testing**	
Target Class	**PACO**	**TD1**	**TD2**
Background	- Background - Dead pixels	- No data - Saturated or defective	- Background
Clear	- Bright - Haze land - Land	- Vegetation - Non vegetated	- Clear land
Atmosphere	- Cirrus cloud - Cirrus cloud thick - Clouds - Medium cirrus land - Medium cirrus water - Saturated - Thick cirrus land - Thick cirrus water - Thin cirrus land - Thin cirrus water	- Cloud medium probability - Cloud high probability - Thin cirrus	- Semi transparent cloud - Cloud
Shadows	- Cloud Shadows - Litopo shadows - Spectral shadows	- Cast shadows - Cloud shadows - Unclassified	- Cloud shadow - Topographic shadow
Water	- Haze water - Water	- Water	- Clear water
Snow	- Snow	- Snow	- Clear snow/ice

Table 3.2: Mapping to common target classes from PACO masking for training and from TD1 and TD2 for testing.

3.4.2 Testing Dataset for the Multispectral Case

This section describes the Testing Datasets (TD) used in this work, including their labeling criteria, observed biases, and sample availability. The model is evaluated using two independently and manually labeled datasets, which were originally developed for external classification evaluation exercises unrelated to PACO or the proposed approach. For simplicity, these are referred to as Testing Dataset # (TD#). The class distribution and pixel counts for each evaluated testing dataset are provided in Table 3.3.

		Pixel Count	
Color	**Target Class**	**TD1**	**TD2**
🟫	Clear	34.93 %	83.86 %
▒	Atmosphere	31.56 %	5.75 %
⬛	Shadows	14.59 %	7.01 %
🟦	Water	15.03 %	2.53 %
⬜	Snow	3.88 %	0.84 %
	Total	1,831,038	1,040,116

Table 3.3: Testing datasets class distribution and pixel count.

Given that the testing datasets belong to other testing and validation exercises (TD1 and TD2, described in Subsections 3.4.2.1 and 3.4.2.2, respectively), a mapping is needed for the common target classes defined for this work. The mapping described in Table 3.2 (columns three and four) ensures consistency between the test datasets and the target classes defined for this work. By standardizing the class definitions, the performance metrics can be uniformly evaluated, allowing a fair comparison of the model's predictions across different datasets. This mapping was empirically defined by analyzing the classes definitions in the testing sets and comparing the results from PACO pre-classification with the available labels for testing. This step is necessary due to the man-made definition of the classes (described in detail in Sections 2.2.4 and 3.4.2.1) and the overlap between them. For example, an analysis of PACO's pre-classification class *cirrus cloud* revealed that it is mapped almost equally to the classes *cloud high probability* and *thin cirrus* from the source set of classes SCL from TD1. Thus, it is impossible to map one class to two without additional information. In this example, all cloud-related classes were merged to create a new class named *atmosphere* (see Table 3.2).

An example of the pixel count matrices used in this study is shown in Table 3.4. Note that some of the split mappings result from well-known problems, such as the misclassification between water and cloud shadows or between snow and cirrus. The missing classes were omitted because they do not have a representation in TD1.

Class	*Cloud shadows*	*Vegetation*	*Non-vegetated*	*Water*	*Cloud medium prob*	*Cloud high prob*	*Thin cirrus*	*Snow*
Cirrus cloud	0.0	0.0	0.0	0.0	3.3	43.6	53.0	0.0
Cirrus cloud thick	0.0	0.0	0.0	0.0	9.1	86.0	4.9	0.0
Cloud shadows	78.3	0.1	2.9	18.4	0.0	0.0	0.2	0.0
Clouds	7.8	0.1	0.1	0.0	2.1	68.4	8.2	13.3
Haze land	4.4	3.4	26.8	6.8	3.1	2.1	53.2	0.0
Haze water	0.1	51.4	0.0	48.3	0.0	0.0	0.0	0.0
Land	7.0	25.6	57.4	0.7	0.1	0.3	8.7	0.0
Medium cirrus land	1.0	0.0	0.0	0.0	2.7	19.2	74.3	2.8
Snow	4.2	0.3	0.0	0.0	0.0	0.9	13.7	80.9
Thick cirrus land	0.0	0.8	0.0	0.0	3.9	21.0	74.2	0.0
Thin cirrus land	0.0	0.8	0.0	0.0	3.9	21.0	74.2	0.0
Water	16.6	2.1	0.0	81.1	0.0	0.0	0.1	0.0

Table 3.4: TD1 pixel count (in percentage by row) for the comparison of class definitions between PACO and Sentinel-2 SCL.

3.4.2.1 TD1: Sentinel-2 Fixed- and Random-sites 2020-2023

This dataset was generated for the Copernicus Sentinel-2 SCL validation activities [122]. A labeling tool was used to generate visually labeled reference pixels in randomized subsets as the validation data. The annotations were based on the visualization of the selected Sentinel-2 Level-2 granule in true- and false-color composites, as well as spectral profiles. The two SWIR bands and the narrow NIR band were selected for the false color composite, along with the L1C cirrus band, to help differentiate the clouds, land and water surfaces.

The scenes were labeled into the following SCL classes [52]: *vegetation*, *non-vegetated*, *water*, *snow*, *topographic* and *cast shadows*, *cloud medium probability*, *cloud high probability*, *thin cirrus*, and *cloud shadow*. The number of pixels per class was manually controlled to be balanced (stratified); however, this depends on the land cover composition of the scenes, as some scenes might not naturally include all the SCL classes.

It comprises 53 manually labeled Sentinel-2 scenes, selected using two different strategies: fixed and randomly selected sites. The fixed site scenes set comprises 40 scenes from the time range of 2020 to 2023, corresponding to the validation exercise of Sentinel-2 [122]. This validation include Germany, Spain, the United States, Russia, and Indonesia. These 5 fixed sites were selected because they are distributed across different continents, covering various climate zones, seasons, and environments. The scene selection is rotated monthly for each site in order to observe seasonality. The chosen monthly products are those with cloud cover closest to 20% and with no-data pixels of less than 33%.

The randomized sites were selected proportional to the number of product requests per continent, comprising 13 annual randomized site scenes from all continents. The Sentinel-2 archive was used to selected the scenes in the Google Earth Engine cloud platform. The first step was to select the time range of the products, then filtered by continent, cloud percentage between 15–25%, snow less than 15%, water less than 20%, and no-data pixels less than 10%. Then random numbers were assigned to each image, and the first image based on these random numbers was selected. A total of three scenes were selected in in Asia, four in Europe, two in North America, one in South America, two in Africa, and one in Australia.

3.4.2.2 TD2: Sentinel-2 High Uncertainty Pixels

This dataset was defined in a previous validation work, also using Sentinel-2 images, presented by Zekoll et al. in [123] to test the Fmask, ATCOR, and Sen2Cor masking algorithms. The tests performed on 20 scenes include a manually labeled reference by a human expert, corresponding to pixels with high uncertainty. The selected pixels for labeling were chosen from those where the three mentioned classifiers disagreed in classification. Therefore, poor performance and low metrics are expected for this dataset, as it does not include average classification cases. Further details and results for this dataset can be found in the aforementioned paper.

3.4.3 Training Dataset for the Hyperspectral Case

In contrast to the multispectral case, where independent manually labeled testing datasets were available for a multi-class classification task, the hyperspectral use case adopts a different validation approach based on physical parameters, as introduced in Section 3.3. Consequently, the training dataset is used to train a DL model for a binary classification task focused on cloud detection.

The training set comprises 55 EnMAP scenes with a wide variety of atmospheric conditions over different types of Earth surfaces. The total number of pixels from all scenes is approximately 90.2 million. However, due to the projection of the scenes, the data is slightly rotated within the distributed EnMAP images, as shown in Figure 3.5, leaving 61.6 million pixels available for training.

Table 3.5 presents the pixel distribution for the binary classification task in the training dataset.

Table 3.5: Training dataset class distribution and pixel count.

Color	**Target Class**	**Pixel Count**
■	No Cloud	79.39 %
▒	Cloud	20.61 %
	Total	61,595,929

Among the cloud-related products available from the Sentinel-5P mission, the CF was identified as a suitable parameter to validate the cloud mask generated by the classifier. Consequently, the masks cube from PACO was mapped according to the criteria presented in Table 3.6, where all atmosphere-related classes were assigned to the *Cloud* class, while the remaining ones were grouped under *No Cloud*.

Since the testing dataset is designed solely to evaluate cloud detection, as previously mentioned in subsection 3.3.2 and described in detail in subsection 3.4.4, the mapping used to generate the training dataset was limited accordingly. Including additional land-cover or atmospheric classes would not provide any added value to this specific evaluation task, as they are not represented or validated in the testing dataset. Therefore, a simplified binary classification approach was adopted to ensure consistency between training and evaluation. However, future work aims to develop a model with class definitions aligned with those in the multispectral case described in subsection 3.4.1. The model itself is not restricted by this limitation and can accommodate any number of classes.

The input dataset for the hyperspectral case follows the same structure as the multispectral case. The spatial dimensions of EnMAP imagery are not fixed but vary between approximately 1,000 and 1,415 rows and columns due to image rotation within the distributed image, generally maintaining a squared proportion factor.

Target Class	**PACO**
Background	- Background - Dead pixels
No Cloud	- Bright - Haze land - Land - Cloud Shadows - Litopo shadows - Spectral shadows - Haze water - Water - Snow
Cloud	- Cirrus cloud - Cirrus cloud thick - Clouds - Medium cirrus land - Medium cirrus water - Saturated - Thick cirrus land - Thick cirrus water - Thin cirrus land - Thin cirrus water

Table 3.6: Mapping to target classes from PACO masking for training in the hyperspectral case.

The input data for the hyperspectral case includes spectral information from the EnMAP mission, a DEM from the Copernicus Mission [53], and an illumination map. All inputs are processed by PACO at 30 m GSD, as previously described in Subsection 3.4.1. For this case, the spectral tensor $\boldsymbol{\mathfrak{X}}_s \in \mathbb{R}^{\text{Rows}\times\text{Columns}\times 224}$, the elevation matrix $\mathbf{D} \in \mathbb{Z}^{\text{Rows}\times\text{Columns}}$, and the illumination map $\mathbf{I} \in \mathbb{R}^{\text{Rows}\times\text{Columns}}$ are used to form the model input.

3.4.4 Testing Dataset for the Hyperspectral Case

The testing dataset is created by identifying matching overpasses between EnMAP and Sentinel-5P. This search is conducted across the entire archive of both missions, from May 2023 to September 2024, focusing on two key criteria: intersection of the captured area and minimal time difference between acquisitions.

Since clouds are constantly moving, the time difference between captures should be as small as possible to minimize the effects of temporal mismatch (see Figure 3.6 and 4.2.3 for details on the temporal thresholds used). As this difference increases, uncertainty in the comparison also grows, influenced by secondary factors that are difficult to quantify. These factors include inherent uncertainties in cloud retrieval from the OCRA algorithm, as discussed in subsection 2.2.6.1, as well as location accuracy and CF retrieval accuracy.

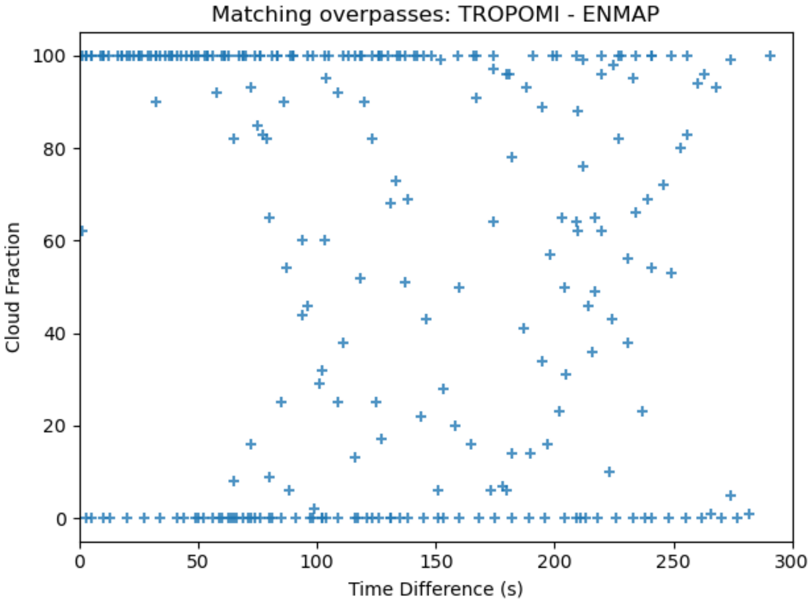


Figure 3.6: Time difference between matching overpasses. CF is retrieved from EnMAP metadata (named cloud coverage).

- Cloud location: The displacement in cloud location depends on cloud velocity, which is influenced by factors such as wind speed and cloud altitude. Even if wind speed can be measured at Earth's surface, it is not uniform across all altitudes. Reducing the time difference between captures minimizes the impact of these secondary factors. Additionally, inherent spatial shifts related to spatial accuracy and geometric corrections contribute to the uncertainty in location matching, which decreases with the CF.

In addition, since captures are not performed at a nadir angle, clouds appear at different locations depending on the variation in sensor viewing angles and cloud altitude [126]. While smaller viewing angle differences and lower cloud altitudes may reduce this displacement, accurate correction is not feasible in this study due to the lack of cloud height information.

- Cloud fraction retrieval accuracy: Due to the approach used by the OCRA algorithm, the accuracy of cloud fraction retrieval decreases when the background is highly reflective or undergoes changes that are not updated in the reference database used for reflectance comparison.

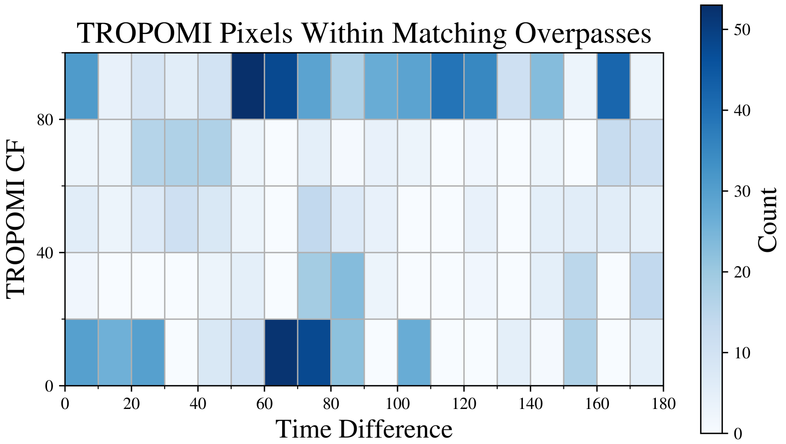
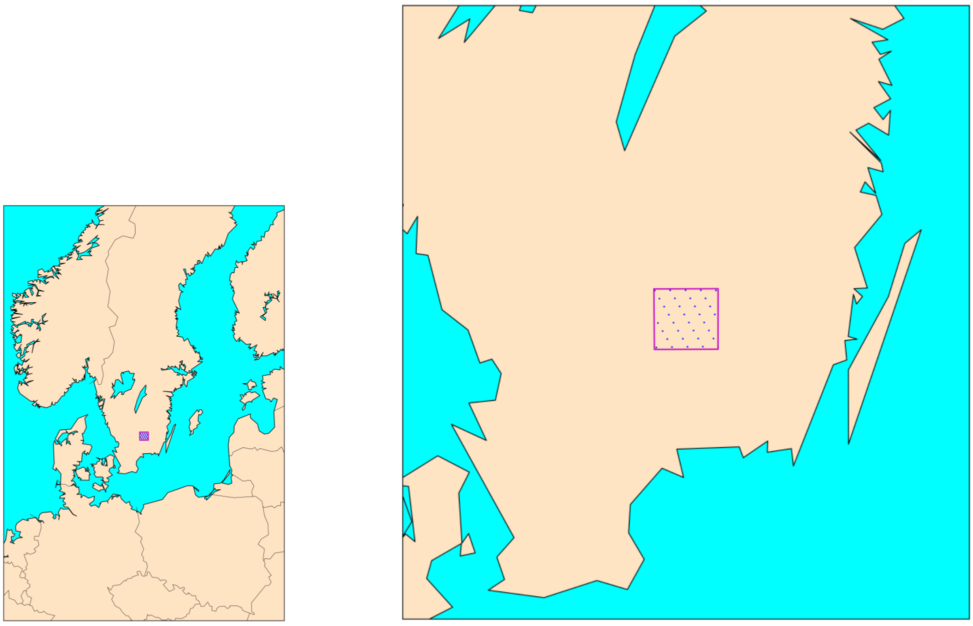


Figure 3.7: Time difference histogram of TROPOMI pixels within EnMAP scenes. Cloud fraction is per pixel from TROPOMI CF product.

The matching TROPOMI pixels with EnMAP scenes are displayed in Figure 3.7. As shown in the plot, the majority of matching pixels with small time differences are concentrated in scenes with either clear skies (0 %) or complete cloud cover (100 %). This distribution is primarily influenced by the acquisition strategy of the EnMAP mission, which is not a systematic monitoring mission but rather task-driven, capturing scenes based on user requests. Consequently, most acquisitions tend to favor cloud-free conditions. However, in regions such as Europe (where the majority of acquisition requests are concentrated) persistent cloudiness is common throughout the year [127], increasing the likelihood of acquiring fully overcast scenes when urgent observations are required despite unfavorable weather conditions.

Given this, all scenes with a capture time difference of less than 180 seconds were included in the testing dataset, resulting in a total of 105 EnMAP scenes with 15 scenes from Sentinel-5P. The time threshold was empirically defined to maximize the number of included scenes and ensure a meaningful statistical representation across different cloud fraction ranges (see Figure 3.7). As the missions continue capturing images, this time threshold could be further reduced.

Given that EnMAP scenes are considerably smaller than the TROPOMI cloud products, only the TROPOMI pixels located within the spatial extent of the EnMAP scenes are retrieved. Figure 3.8 illustrates a matching overpass, where the magenta square represents the EnMAP scene and the blue dots indicate the centers of the corresponding TROPOMI pixels. This selection is performed by comparing the geographic coordinates (latitude and longitude) of the TROPOMI pixel centers with the corner coordinates of the EnMAP scene.



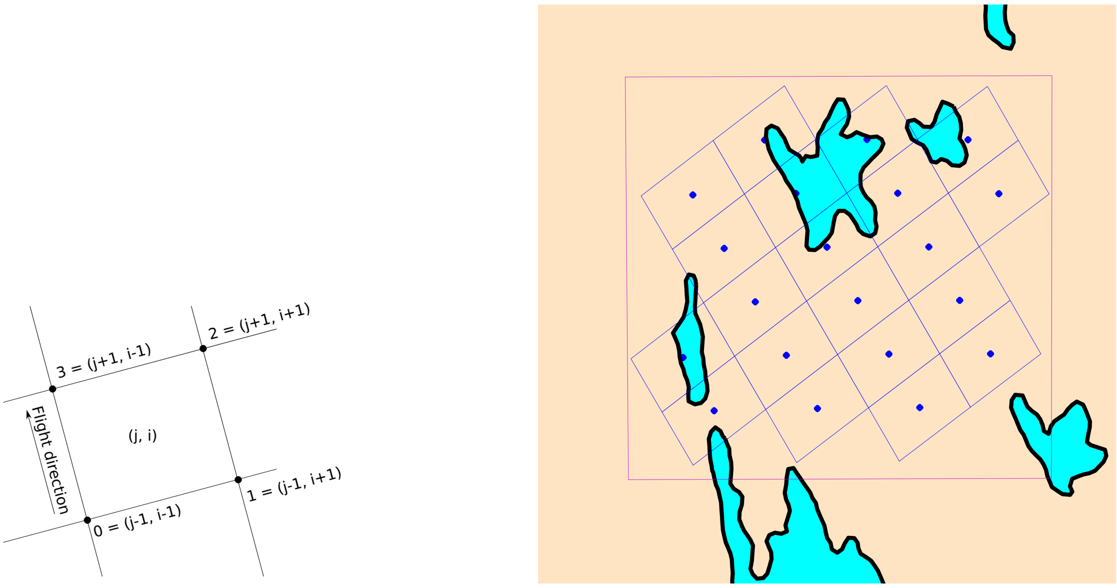
(a) Location over Sweden.

(b) Zoom in.

Figure 3.8: Matching overpass between EnMAP and Sentinel-5P. Magenta outlines denote the boundaries of the EnMAP scene, while blue dots mark the centers of TROPOMI pixels.

Once the TROPOMI pixel centers within the EnMAP scene are identified, the entire area of each TROPOMI pixel must be contained within the EnMAP image, as cloud fraction is computed using the EnMAP pixels enclosed by a TROPOMI pixel for comparison.

If any portion of a TROPOMI pixel extends beyond the boundaries of the EnMAP image, it must be discarded. The geolocation of pixel centers and corners is provided in the TROPOMI metadata [55]. In Figure 3.9a, a representation of how the TROPOMI metadata defines pixel corners is displayed, while Figure 3.9b illustrates the outlines of TROPOMI pixels.



(a) Geolocation of TROPOMI pixel centers and corners, included in metadata of cloud products. Plot credit: L2-Algorithm Theoretical Basis Document (ATBD) TROPOMI [55].

(b) TROPOMI pixels contained within the EnMAP scene. The magenta outlines indicate the boundaries of the EnMAP scene, while the blue dots represent the pixel centers, and the blue lines denote the outlines of TROPOMI pixels.

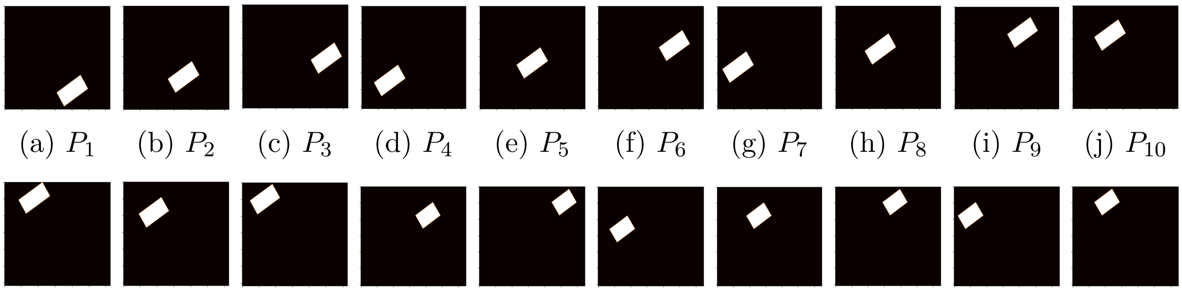
Figure 3.9: TROPOMI pixel geolocation within an EnMAP scene.

After obtaining the geolocated corners of TROPOMI pixels over the EnMAP scenes, rectangles are drawn onto binary masks using linear regression to determine the incident lines between pairs of adjacent corners. These lines are mapped onto the binary mask corresponding to the EnMAP scene, and the pixels within the four sides of each rectangle are then filled. These masks serve as filters to extract the EnMAP pixels contained within each TROPOMI pixel. In Figure 3.10, the binary masks corresponding each of the i^{th} pixel displayed (denoted as P_i) in Figure 3.9b are shown.

As a last step, the EnMAP CF to evaluate is computed from the cloud mask of the EnMAP scene using equation (3.14) for each of the i^{th} pixels of TROPOMI.

$$CF_i^{\text{EnMAP}} = \frac{p_{\text{cloud}}}{p_{\text{total}}}(100\%) \tag{3.14}$$

where: p_{total} = Number of EnMAP pixels inside a TROPOMI pixel.
p_{cloud} = Number of Cloud EnMAP pixels inside a TROPOMI pixel.



(a) P_1 (b) P_2 (c) P_3 (d) P_4 (e) P_5 (f) P_6 (g) P_7 (h) P_8 (i) P_9 (j) P_{10}

(k) P_{11} (l) P_{12} (m) P_{13} (n) P_{14} (o) P_{15} (p) P_{16} (q) P_{17} (r) P_{18} (s) P_{19} (t) P_{20}

Figure 3.10: Binary masks of TROPOMI pixels within the EnMAP scene, illustrating 20 different pixels.

Figure 3.11 shows, an example of an EnMAP cloud mask (GSD: 30 m) and the resulting CF, adjusted to match the location and spatial resolution of TROPOMI pixels (GSD: 5.5 km × 3.5 km).

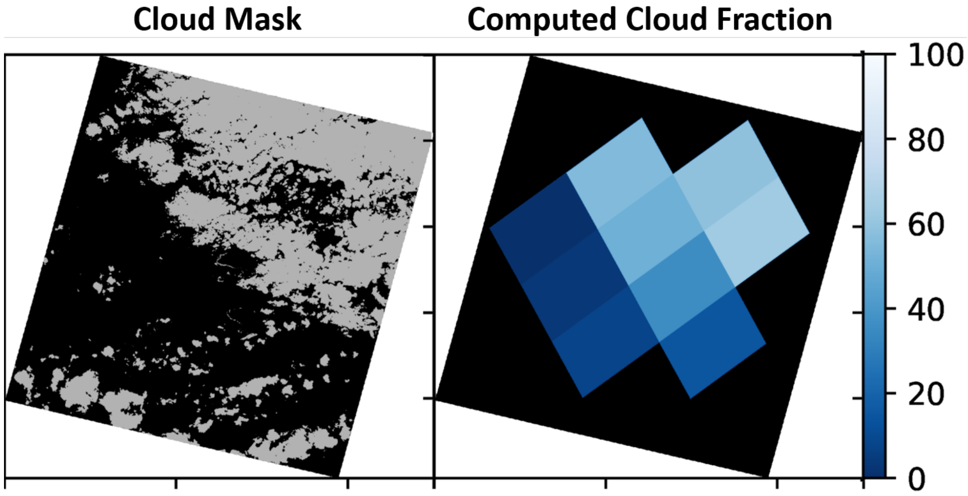


Figure 3.11: EnMAP cloud mask (left plot) is transformed into CF (right plot), matching the spatial resolution and location of TROPOMI pixels.

According to the OCRA algorithm validation [128], this method of calculating cloud fraction is known as *geometric cloud fraction*, as it is derived from a higher spatial resolution source. In contrast, the cloud fraction retrieved with TROPOMI is referred to as *radiometric cloud fraction*, since it relies solely on sensor measurements.

Geometric cloud fraction is expected to overestimate the actual CF [56]. This occurs because the radiometric approach assigns lower weight to clouds with low reflectance. For instance, when a high-altitude cloud casts a shadow over neighboring clouds, thereby, reducing the total cloud fraction accounted for. In contrast, geometric cloud fraction does not incorporate radiometric variations and instead classifies pixels as either cloudy or non-cloudy, leading to a systematic discrepancy.

After applying this process to all available overlapping overpasses available from May 2023 until September 2024, a total of 982 TROPOMI pixels were obtained matching EnMAP for use as reference data in the testing dataset for the hyperspectral case.

3.5 Pixel Selection

For each of the use cases, the first step in the training process (red-dashed box in Figure 3.1) is the “Pixel Selection”, where high-quality pixels are chosen to train the model. Since the quality of the training data has a strong impact on the generalization capacity of deep learning models, this step aims to reduce label noise on training datasets to improve performance of DL models.

This work analyzes the model’s behavior when trained using three pixel selection strategies taking or not into account the confusion present in the original PACO masking cube ($\mathcal{M}$).

- *No filter*: using all the pixels, regardless of the uniqueness of the classification in the previous PACO output, as described in Section 3.5.1.
- *Uniclass filter*: selecting pixels classified into a unique class in the original $\mathcal{M}$ masks cube from PACO, described in 3.5.2.
- *Physics filter*: selecting pixels based on the physics rules defined in Table 3.7, described in 3.5.3.

The proposed pixel selection strategies leverage the multi-label classification from spectral indices thresholding ($\mathcal{M}$), allowing classification of a pixel into one or more pre-classification classes. Furthermore, this characteristic can be used to identify pixels with high classification uncertainty when the active masks are contradictory, e.g., when a pixel is classified as snow and thick clouds. These pixel selection strategies for composing training datasets are described in the following subsections.

3.5.1 Baseline - No Filter

For baseline comparison, a training set was defined using all the available pixels to evaluate how our pixel selection strategy performs relative to the complete dataset. This filter is referred to as “*No filter*” in the results section.

3.5.2 Unique Class Filter

This filter was incorporated as a conservative approach for composing the training dataset, that selects only pixels classified into a single class. In this context, the greater the number of active masks in a pixel, the higher the uncertainty in its class assignment. Therefore, to ensure that only reliable labels are used during training, this method selects only pixels classified into a single class. For example, a Sentinel-2 image and its corresponding active mask map are shown in Figure 3.12.

In the image displayed in Figure 3.12, pixels with a value of 1 in the right plot are retained. In contrast, some pixels are discarded because, although the water mask is active, they also have values corresponding to snow/ice (i.e., 2 and 3.), indicating overlapping or ambiguous classifications.

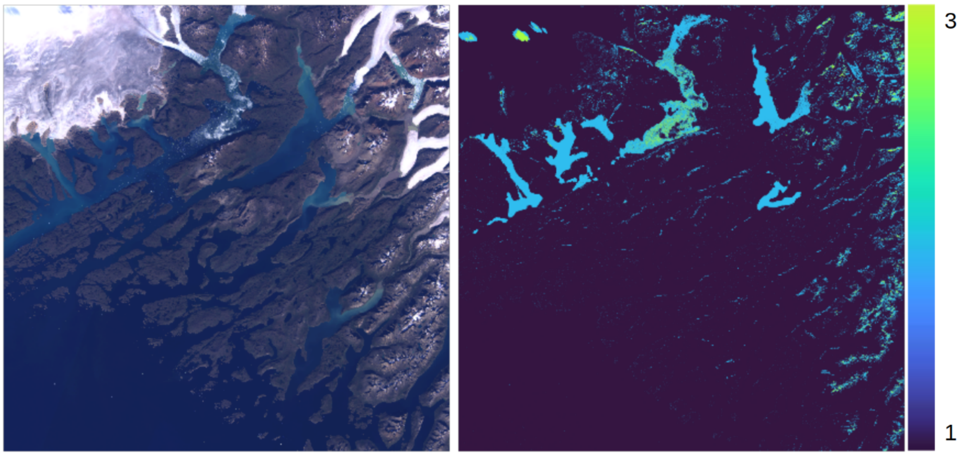


Figure 3.12: Example of active masks in a Sentinel-2 image (right). The RGB image on the left serves as a reference. Image: Sentinel-2, T23VMH, 2017/09/04.

This filtering method provides labels with lower uncertainty than the original masks cube ($\mathcal{M}$) at the expense of under-representing certain classes. As the classification remains conservative, borderline cases (higher uncertainty classification cases) are left for the neural network to handle. This filter is referred to as the *Uniclass filter* in the results section.

3.5.3 Filtering Based on Physics Rules

In addition to the proposed pixel selection criteria, a pixel selection strategy was defined based on physics-based rules, as presented in Table 3.7. In this context, physics-based refers to grouping classes according to their physical compatibility, particularly in terms of expected land surface and atmospheric properties. Pixels exhibiting class combinations that are physically inconsistent or contradictory (e.g., such as pixels labeled simultaneously as both snow and water as illustrated in Figure 3.12, or the co-occurrence of thick clouds and clear-sky land in the same pixel) are excluded from the training set. These contradictions are indicated by gray cells in Table 3.7. By removing such ambiguous cases, the model is trained on physically coherent examples, enabling it to infer class confidence from representative patterns in spatial and spectral contexts. At the same time, this strategy preserves valuable information from valid multi-label cases, where a pixel legitimately belongs to more than one compatible class. This filter is referred to as the "*Physics filter*" in the results section.

Table 3.7 shows the rules for pixel selection used in this work. These rules were defined by grouping compatible classes and identifying physically contradictory principles, based on the principles of radiative transfer and the expected spectral behavior of surfaces and clouds. The rules also reflect the current implementation and definitions used in the PACO masking algorithm, such as the presence of non-optically opaque clouds or cloud shadows over different surfaces. Note that in some cases, even if the spectral information does not allow the detection of certain elements, e.g., cloud shadows under a cloud, shadows physically exist and can be masked from the projection of the clouds. Another example is the topographic shadow under clouds; even if the cloud is blocking the view, the topographic shadow is computed from the DEM.

Figure 3.13 shows an example of the performed pixel selection, where pixels classified as snow and clouds (a gray cell in Table 3.7) are discarded by this filter from the training datasets by assigning them the background label. In contrast, the pixels of snow/ice in the river are included, as they pass the filter (green cells in Table 3.7).

Group	**Surface**							**Low Atmosphere (< 3-4 km)**			**High Atmosphere (> 3-4 km)**								**Hardware**		
CLASS	**Land**	**Water**	**Snow**	**Bright**	**Cloud shadows**	**Spectral shadows**	**Litopo shadows**	**Haze land**	**Haze water**	**Clouds**	**Thin cirrus land**	**Thin cirrus water**	**Medium cirrus land**	**Medium cirrus water**	**Thick cirrus land**	**Thick cirrus water**	**Cirrus cloud**	**Cirrus cloud thick**	**Background**	**Saturated**	**Dead pixels**
Land	x	⬛	⬛	⬛	🟩	🟩	🟩	🟩	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛
Water	⬛	x	⬛	⬛	🟩	🟩	🟩	⬛	🟩	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	⬛
Snow	⬛	⬛	x	⬛	🟩	🟩	🟩	🟩	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛
Bright	⬛	⬛	⬛	x	🟩	🟩	🟩	🟩	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	🟩	⬛
Cloud shadows	🟩	🟩	🟩	🟩	x	🟩	🟩	🟩	🟩	⬛	🟩	🟩	🟩	🟩	🟩	🟩	⬛	⬛	⬛	⬛	⬛
Spectral shadows	🟩	🟩	🟩	🟩	🟩	x	🟩	🟩	🟩	⬛	🟩	🟩	🟩	🟩	🟩	🟩	⬛	⬛	⬛	⬛	⬛
Litopo shadows	🟩	🟩	🟩	🟩	🟩	🟩	x	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	⬛	🟩	🟩
Haze land	🟩	⬛	🟩	🟩	🟩	🟩	🟩	x	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛
Haze water	⬛	🟩	⬛	⬛	🟩	🟩	🟩	⬛	x	⬛	⬛	🟩	⬛	🟩	⬛	🟩	⬛	⬛	⬛	⬛	⬛
Clouds	⬛	⬛	⬛	⬛	⬛	⬛	🟩	⬛	⬛	x	🟩	🟩	🟩	🟩	🟩	🟩	⬛	⬛	⬛	🟩	⬛
Thin cirrus land	🟩	⬛	🟩	🟩	🟩	🟩	🟩	🟩	⬛	🟩	x	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛
Thin cirrus water	⬛	🟩	⬛	⬛	🟩	🟩	🟩	⬛	🟩	🟩	⬛	x	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛
Medium cirrus land	🟩	⬛	🟩	🟩	🟩	🟩	🟩	🟩	⬛	🟩	⬛	⬛	x	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛
Medium cirrus water	⬛	🟩	⬛	⬛	🟩	🟩	🟩	⬛	🟩	🟩	⬛	⬛	⬛	x	⬛	⬛	⬛	⬛	⬛	⬛	⬛
Thick cirrus land	🟩	⬛	🟩	🟩	🟩	🟩	🟩	🟩	⬛	🟩	⬛	⬛	⬛	⬛	x	⬛	⬛	⬛	⬛	🟩	⬛
Thick cirrus water	⬛	🟩	⬛	⬛	🟩	🟩	🟩	⬛	🟩	🟩	⬛	⬛	⬛	⬛	⬛	x	⬛	⬛	⬛	🟩	⬛
Cirrus cloud	⬛	⬛	⬛	⬛	⬛	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	x	⬛	⬛	🟩	⬛
Cirrus cloud thick	⬛	⬛	⬛	⬛	⬛	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	x	⬛	🟩	⬛
Background	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	x	⬛	⬛
Saturated	⬛	⬛	⬛	🟩	⬛	⬛	🟩	⬛	⬛	🟩	⬛	⬛	⬛	⬛	🟩	🟩	🟩	🟩	⬛	x	⬛
Dead pixels	⬛	⬛	⬛	⬛	⬛	⬛	🟩	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	x

Table 3.7: Physics rules for pixel filtering. Pixels classified as coincident green cells are included in the training set, while gray cells are discarded.

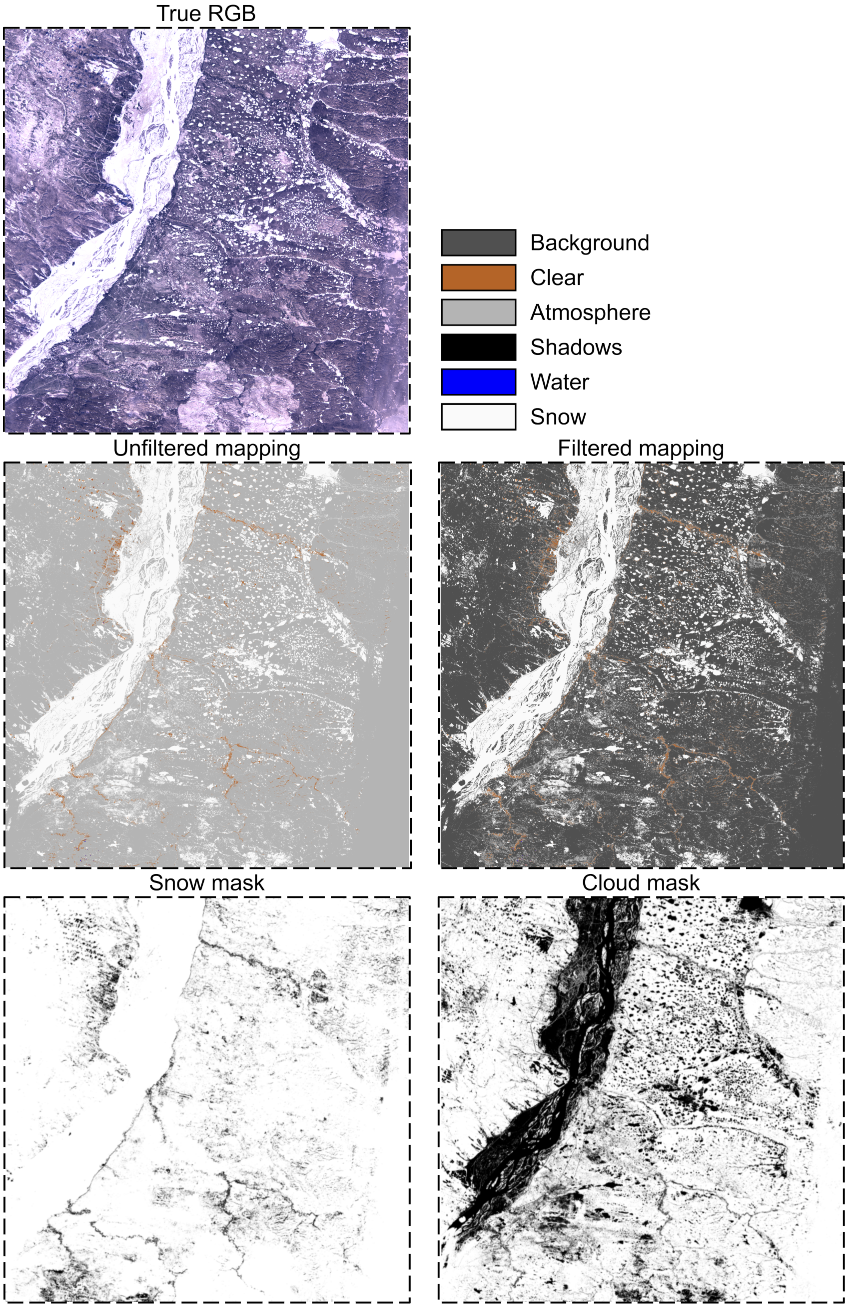


Figure 3.13: Example of pixel selection for training. The binary masks at the bottom, from PACO's masks cube, indicate contradictory classifications of snow and clouds. Pixels with both masks active are discarded from training by assigning them the background label. A comparison of unfiltered and filtered mapping is shown in the two middle images. The RGB image at the top serves as a reference. Image: Yakutsk, Russia. Sentinel-2, T52VEP, 2023/03/17, 11:45, -13°C.

3.6 Deep Learning-Based Classifier

DL models, particularly those based on CNNs, have demonstrated outstanding performance in image classification by effectively capturing characteristic spatial features for object identification [129, 130, 131]. On the other hand, remote sensing optical imagery, which consists of highly spatially correlated pixels, also incorporates spectral information. Characteristic patterns in the spectrum enable the classification of elements on the Earth's surface or in the atmosphere. Given the high volumes of data retrieved daily from EO missions, this challenge is a suitable candidate for the implementation of DL models.

The approach implemented in this thesis involves training a DL model using the multi-label classification outputs generated by spectral index-based masking algorithms, such as those implemented in PACO or similar atmospheric correction tools. This strategy enables the efficient compilation of large-scale training datasets without requiring manual annotation, which is often unfeasible at a global scale due to the complexity and resource demands of producing high-quality reference labels. By relying on algorithmic pre-classification masks, the resulting labels remain consistent with physically-informed class definitions, grounded in prior knowledge from radiative transfer theory and spectroscopy. Although this work focuses on a specific model architectures based on CNNs, the proposed methodology is designed to be adaptable and may serve as a framework for the implementation and evaluation of emerging DL models in operational remote sensing applications.

In this case, where pixels are discarded as explained in subsection 3.5, the model processes input pixel by pixel and in the case of CNN2D and CNN3D, along with its neighboring pixels, forming spatial-spectral patches. Figure 3.14 illustrates the structure of these patches.

A key advantage of this input representation for the DL model is that, even if the neighboring pixels included in the patch are not labeled or used directly for training, they still contribute to the spatial context to the labeled pixels. For example, at the boundary between cloud and snow pixels, both classes may be active, leading to their exclusion. However, these pixels remain within the spatial neighborhood of low-uncertainty pixels, providing useful information to the model of the spatial context of a pixel.

As mentioned in the state-of-the-art review (subsection 2.4.3), a set of CNNs with variant feature extractors were selected due to its relatively low computational complexity and proven outstanding performance [68]. The architectures of the models used in this thesis are shown in Table 3.8. Originally presented in [68] for hyperspectral imagery, the model was adapted for the multispectral case by reducing the size of the filters that perform convolutions in the spectral domain. For the hyperspectral case, the reduced architecture presented in this thesis was retained, further lowering computational complexity.

For the methodology proposed in this thesis, a window size of 11 × 11 pixels was

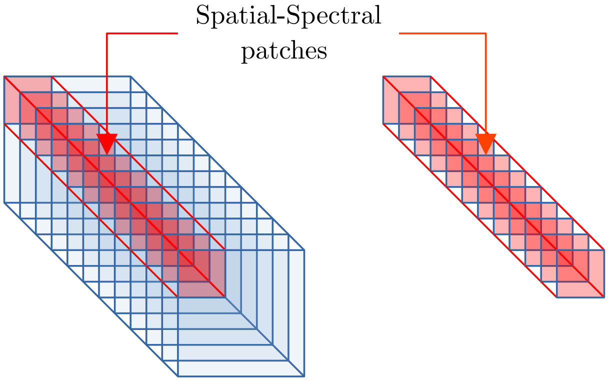


Figure 3.14: Input of the implemented DL model. Spatial-spectral patches with the pixel of interest at the center. The blue cube represents a multi/hyperspectral image $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{\text{Rows}\times\text{Columns}\times F}$, while the red cube denotes $\boldsymbol{\mathcal{X}}_{\text{patch}} \in \mathbb{R}^{W\times W\times F}$, where W is the window size and F is the number of features, defining the input of the CNN2D and CNN3D models.

selected, resulting in an input tensor $\boldsymbol{\mathcal{X}}_{\text{patch}} \in \mathbb{R}^{11\times 11\times F}$ centered at $(6, 6)$. This size provides enough spatial context (220 m for Sentinel-2 and 330 m for EnMAP) to capture spatial patterns to characterize the observed objects on Earth surfaces. Larger window sizes were tested empirically, but no significant performance improvements were observed, while computational complexity increased. In CNN models, increasing the window size proportionally increases the number of convolutions required for each new pixel (assuming unit stride) at each layer.

Each model performs feature extraction in different domains. CNN1D analyzes local relationships among neighboring wavelengths by performing 1D convolutions with kernels of size $(1 \times F)$. Therefore, the feature extraction only depends on the spectral signature for each pixel. CNN2D first extracts spatial features for each band independently with kernels of size (5×5). These band-specific feature maps are then passed to the detection stage, where spectral relationships across bands are jointly modeled. Similarly, CNN3D extracts spatial and spectral features jointly with 3D convolutions using kernels of size $(5\times 5\times 10)$, extracting complex relationships between the spatial and spectral information and then passed to a detection stage.

In subsequent layers of the three architectures mentioned, the spectral, spatial, and spatial-spectral features are progressively reduced and passed through a fully connected layer, which assigns confidence indices at the output of the CNNs using a softmax activation function [68].

In order to include the information from the multi-label mask cube, the softmax activation function $f(\mathbf{s})_c$ is used for each class $c \in 1, 2, \cdots, C$, as defined in equation (3.15), in conjunction with the categorical Cross-Entropy (CE) loss, denoted as $CE(\mathbf{t}, \mathbf{s})$,

Model	**Main Layer**	**Norm**	**Activation function**	**Down sampling**
CNN1D	Linear input $(1 \times F)$	-	-	-
	Conv1D(50×15)	-	ReLU	-
	Dense(100)	BN	ReLU	-
	Dense(n_class)	-	Softmax	-
CNN2D	Linear input $(11 \times 11 \times F)$	-	-	-
	Conv2D$(50 \times 5 \times 5)$	-	ReLU	-
	Conv2D$(100 \times 5 \times 5)$	-	ReLU	POOL(2×2)
	Dense(100)	-	ReLU	-
	Dense(n_class)	-	Softmax	-
CNN3D	Linear input $(11 \times 11 \times F)$	-	-	
	Conv3D$(32 \times 5 \times 5 \times 10)$	BN	ReLU	-
	Conv3D$(64 \times 5 \times 5 \times 5)$	BN	ReLU	POOL$(2 \times 2 \times 1)$
	Dense(300)	BN	ReLU	-
	Dense(n_class)	-	Softmax	-

Table 3.8: CNN1D, CNN2D, and CNN3D model descriptions [68].

defined in equation (3.16):

$$f(\mathbf{s})_c = \frac{e^{s_c}}{\sum_i^C e^{s_i}}, \tag{3.15}$$

$$CE(\mathbf{t}, \mathbf{s}) = -\sum_c^C t_c \cdot log\left(f(\mathbf{s})_c\right), \tag{3.16}$$

where $\mathbf{s}$ is the vector containing the scores s_c inferred at the output of the model, and $\mathbf{t}$ is the reference (also called truth) multi-label vector. This loss function compares the confidence index at the output of the softmax activation functions, $[f(\mathbf{s})_1, f(\mathbf{s})_2, \cdots, f(\mathbf{s})_C]$, with the multi-label classification used as a reference for training. A confidence index vector is obtained from the one-hot encoded classes by dividing each binary value t_c^b (casted to float) by the number of active masks, such that:

$$\mathbf{t} = \frac{\left[t_1^b, t_2^b, \cdots, t_C^b\right]}{\sum_{i=1}^C t_i^b}, \tag{3.17}$$

in this way, the multi-label reference matches the confidence index format of the softmax function by assigning equal values to each class. Future work will explore assigning the confidence index of the reference multi-labels using information from physics-based models and investigating the behavior with other activation functions. This is motivated by the fact that softmax, as a generalization of the sigmoid activation function, is widely used for binary classification but tends to polarize the confidence index [132]. Lastly, to address the naturally imbalanced representation of each class in the training set, a weighting scheme is applied for each class as follows [133]:

$$w_c = \frac{|\mathbb{P}|}{C \cdot |\mathbb{P}_c|}, \tag{3.18}$$

where $|\mathbb{P}|$ is the cardinality of the set of pixels and $\mathbb{P}_c$ the set of pixels of class c for training. To obtain $|\mathbb{P}_c|$, a mapping is performed from the multi-label mask cube to a single-label classification map using a fixed priority hierarchy in the following order: *Atmosphere*, *Shadows*, *Snow*, *Water*, and *Clear*. The obtained pixel count provides an approximate class distribution for computing loss weights. Hence, the loss function leverages the multi-label classification from threshold-based classifiers, as it compares the confidence index of a pixel belonging to a class with the confidence indices of other classes, which enables a more informed classification criterion for mixed pixels, such as cloud edges, or for pixels with similar spectral signatures, such as snow and clouds.

3.7 Training

Global-scale applications require a high level of generalization, which is achieved by preventing overfitting during the model fitting process. This is particularly crucial in this thesis, as the testing dataset consists of scenes not used during training, and the corresponding labels originate from independent sources.

Training is the most computationally intensive phase of neural network implementations and is highly dependent on the Central Processing Unit (CPU), Random Access Memory (RAM), and Graphics Processing Unit (GPU) available. GPU computing performance depends on the available GPU memory, Compute Unified Device Architecture (CUDA) cores, memory bandwidth, associated accelerators such as tensor cores for tensor multiplication, and the architecture itself. Therefore, training was performed using High Performance Computing (HPC) centers, highlighting below some of the key characteristics. During the early stages of this thesis, we used the HPC in Cinvestav Guadalajara, featuring ×1 Nvidia Tesla V100 16 GB. Then, we used the Othrys HPC, featuring ×16 Nvidia GeForce RTX 2080Ti 11 GB, ×8 Nvidia TITAN RTX 24 GB, ×2 Nvidia Quadro P6000 24 GB, and ×12 Nvidia Titan X 12 GB. During the final phase of this thesis, we used the HPC Terrabyte of LRZ, which has 45 nodes, each with ×2 CPU 40-core Intel Xeon, 1024 GB of RAM, and ×4 GPU Nvidia A100 80 GB.

3.7.1 Training the Model for the Multispectral Case

The results presented in this work for the multispectral case are from models trained with the Adam optimization algorithm [134], using 1 % (approximately 20 million pixels) of the available 71 Sentinel-2 scenes with 15 features (13 spectral bands, DEM, and illumination), and a batch size of 100,000 samples. This small fraction is sufficient to train the model effectively due to the high spatial redundancy in remote sensing imagery and the random selection of pixels across diverse locations. These fitting hyperparameters (including learning rate, batch size, optimizer, train size) were selected based on an exhaustive grid search evaluating the model's performance across different parameter combinations.

After applying the pixel selection strategies described in Section 3.5, two subsets are selected: one for training and one for validation. Each subset is selected with a random-stratified strategy, maintaining the same proportion of pixels for each class. The objective of including a validation set is to avoid over-fitting by evaluating the model at the end of each epoch, and discarding any further changes if the validation evaluation does not improve.

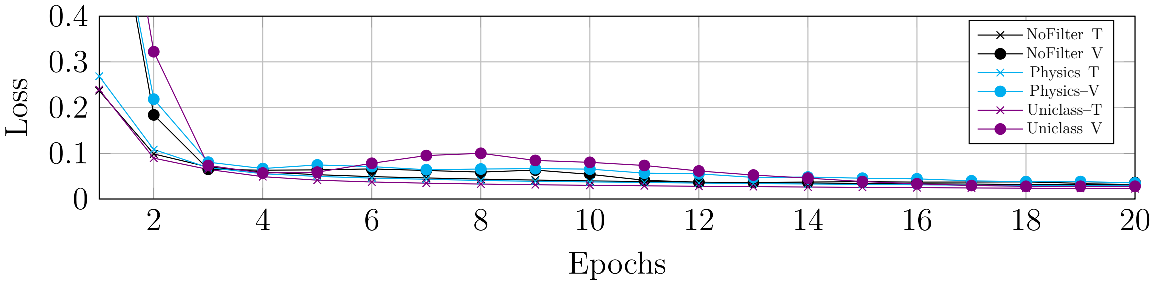
The validation set consists of 1 % ($\approx$ 20 million pixels) of the available data for training and is fixed for consistent comparison across trainings. The loss curves obtained by training for the three pixel selection strategies, using the hyperparameters previously mentioned, are shown in Fig. 3.15a, 3.15b, and 3.15c, for CNN1D, CNN2D, and CNN3D, respectively.

Table 3.9 reports the parameters counts and wall-clock times measured using a GPU NVIDIA A100 (80GB) with a 12-core CPU and 256GB of RAM. For CNN2D and CNN3D, the spatial patches require extensive use of memory (with size $11 \times 11 \times 15$ per pixel), so a custom batch generator streams the data during both training and inference. All Sentinel-2 scenes were resampled to 20 m spatial resolution; times therefore reflect the operational cost at that resolution.

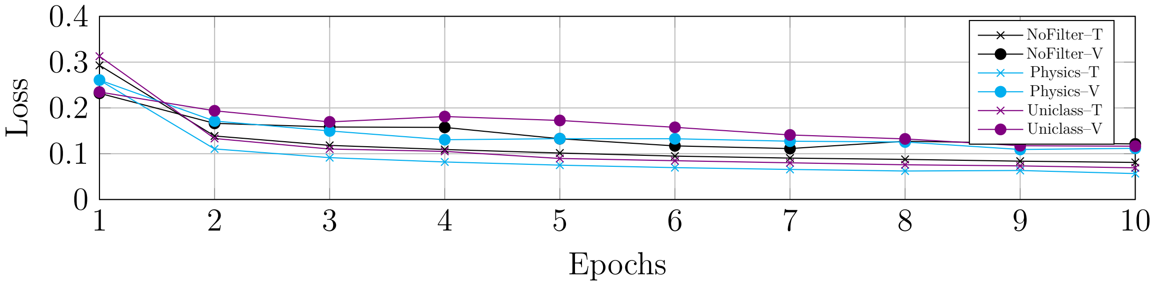
Table 3.9: CNNs size and time complexity training for the multispectral case.

Model	**Parameters**	**Epochs**	**Training time (min)**	**Prediction time (min)**
CNN1D	76,805	20	161.9	0.6
CNN2D	154,505	10	162.0	3.0
CNN3D	305,885	30	482.9	3.0

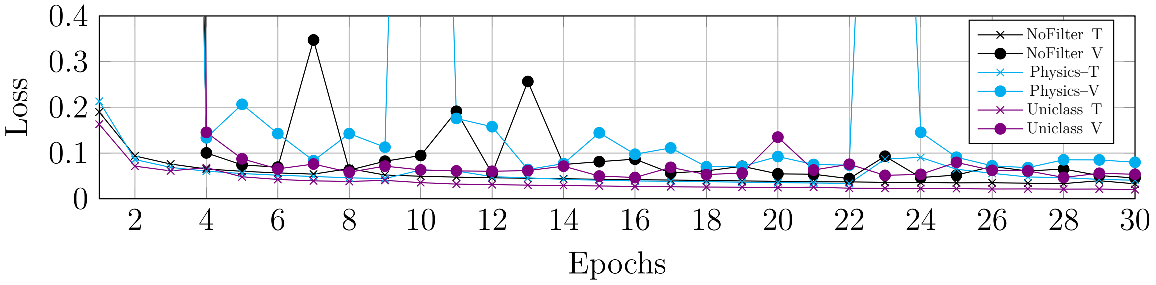
To avoid introducing any information from the testing dataset into the training process and to explore methods for obtaining a generalized model applicable to any case, an analysis revealed that increasing the number of training pixels or training epochs (despite slight improvements in the validation set with each epoch) does not lead to better performance for the testing datasets. This behavior is likely due to the fact that increasing the size of the training set also increases the likelihood of introducing outliers. Additionally, by introducing highly correlated pixels corresponding to average cases, the



(a) CNN1D



(b) CNN2D



(c) CNN3D

Figure 3.15: Training (T) and validation (V) loss curves for the three CNN models under the three pixel-selection strategies for the multispectral case. The model saved during training corresponds to the epoch with the lowest validation loss, evaluated at each epoch.

training process may overfit the original model, replicating its behavior and losing the generalization that comes from a sparse input. However, these observations remain hypothetical and should be further investigated. More details on this topic are discussed in Subsection 4.3.3.

3.7.2 Training the Model for the Hyperspectral Case

For the hyperspectral case, using the same optimizer and architecture backbone as in the multispectral case (keeping the same size of the filters but modifying the linear input from $(11 \times 11 \times 15)$ to $(11 \times 11 \times 226)$. The models were trained during 10 epochs using 10 % ($\approx$ 12 million pixels) of the available 55 EnMAP scenes after applying the pixel selection strategies described in Section 3.5. The use of 10% was found to be a good balance between model generalization and available computational resources, considering the higher dimensionality of hyperspectral imagery, with 226 features (224 spectral bands, DEM and illumination) and a batch size of 5000 samples.

The validation set consists of 10 % ($\approx$ 12 million pixels) of the available data for training and is fixed for consistent comparison across trainings.

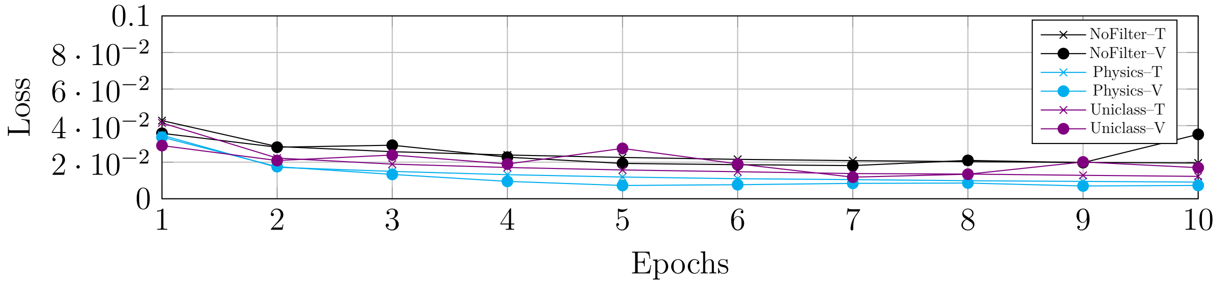
Table 3.10 reports the parameters counts and wall-clock times measured using a GPU NVIDIA A100 (80GB) with a 12-core CPU and 256GB of RAM with the previously mentioned custom batch generator for spatial patches of size $11 \times 11 \times 226$ per pixel. All EnMAP scenes have a 30 m spatial resolution; times therefore reflect the operational cost at that resolution. It can be observed that the parameters for CNN1D and CNN3D (models that extract spectral features) increase significantly the number of parameters to train, this is due to the significant increase of features in the spectral domain (from 15 to 226, a ×15 increase), therefore the number of features and operations within layers increase accordingly. On the other hand, CNN2D only present a linear increase in the amount of spatial features (in the first layer, one for each new band), as well in the number of connections on the classification stage (last layers), where all spatial maps are flattened in the fully connected layer.

Table 3.10: CNNs size and time complexity training for the hyperspectral case.

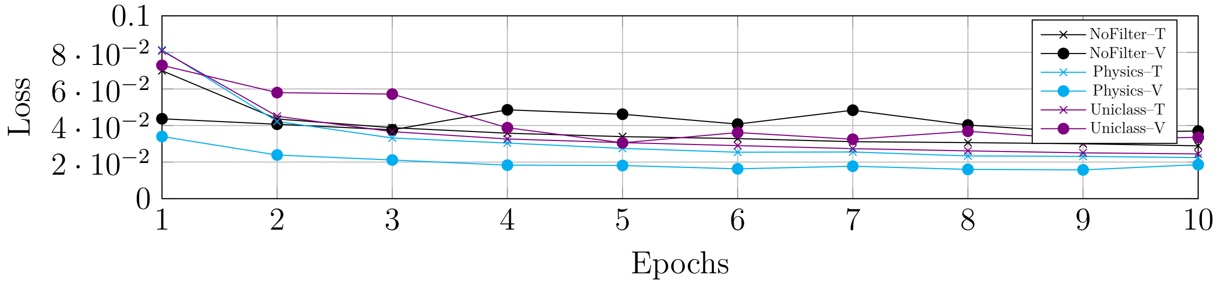
Model	**Parameters**	**Epochs**	**Training time (min)**	**Prediction time (min)**
CNN1D	1,142,052	10	8.58	1.2
CNN2D	417,952	10	179.2	1.8
CNN3D	4,356,182	10	520.5	1.8

The loss curves obtained by training for the three pixel selection strategies presented using the hyperparameters previously mentioned are shown in Figure 3.16.

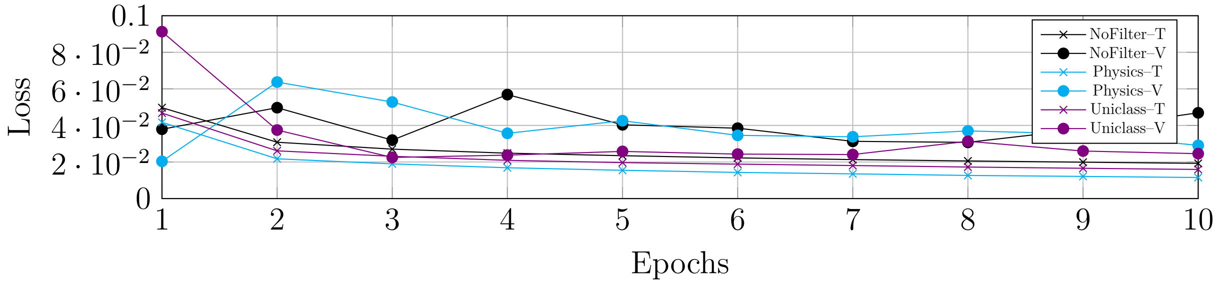
Although the training procedure is consistent across the multispectral and hyperspectral cases (with equivalent model architectures, patch sizes, and pixel selection strategies) the resulting loss curves show notable differences, as seen in Figures 3.15 and 3.16. In the multispectral case, the models show generally higher loss, particularly at the beginning of the training and slightly noisier validation curves, particularly in CNN3D, possibly due to the limited number of spectral features and the more diverse class set used in training



(a) CNN1D



(b) CNN2D



(c) CNN3D

Figure 3.16: Training (T) and validation (V) loss curves for the three CNN models under the three pixel-selection strategies for the hyperspectral case. The model saved during training corresponds to the epoch with the lowest validation loss, evaluated at each epoch.

(5 classes for the multispectral case, 2 classes for the hyperspectral case). In contrast, the hyperspectral training curves are smoother and converge more rapidly across all CNN architectures. This is likely due to the binary classification setup and the increased spectral resolution of EnMAP, which provides richer spectral signatures that improve class separability. However, this increased dimensionality also introduces challenges related to the curse of dimensionality (such as sparsity, distance distortion, and risk of overfitting) as discussed in Section 2.3.1.2.

3.8 Testing

Once the models are fitted, the testing phase must be conducted using different scenes and independent sources of information. This step is necessary to align with the global implementation perspective of this work and to demonstrate the generalization ability and limitations of the proposed approach. To evaluate this for the multispectral, multi-class classification testing sets are compared with the model's predictions using classification performance metrics, analyzing the bias of each while considering the imbalanced availability of samples of each class. On the other hand for the hyperspectral case, correlation metrics are used to evaluate a linear regression evaluating the CF.

3.8.1 Classification Metrics

Given a confusion matrix $\mathbf{C}$ of K classes:

$$\mathbf{C} \in \mathbb{Z}^{K\times K}, \tag{3.19}$$

such that:

$$\mathbf{C} = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,K} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ c_{K,1} & c_{K,2} & \cdots & c_{K,K} \end{bmatrix} \quad \text{e.g., for } K=2,\ \mathbf{C} = \begin{bmatrix} \mathrm{TP} & \mathrm{FN} \\ \mathrm{FP} & \mathrm{TN} \end{bmatrix}, \tag{3.20}$$

where columns and rows correspond to the reference and predicted labels respectively. When $K = 2$, TP, TN, FP, and FN are the true positives, true negatives, false positives, and false negatives, respectively.

The confusion matrix provides a general overview of correctly classified pixels (in diagonal), such that $\{c_{i,j} | i = j\}$, and misclassifications with other classes $\{c_{i,j} | i \neq j\}$, while allowing the computation of any needed metric for comparison.

In addition, most of the results are presented in terms of Matthew's Correlation Coefficient (MCC) for the general evaluation and comparison between the baseline and the prediction obtained. It is used for the multi-class case (as a single-score metric for each testing dataset) and also for class-wise comparisons. However, also the inclusion of precision and recall metrics allows for an intuitive interpretation of overestimation or underestimation of a class.

To individually evaluate each class, the multi-class confusion matrix is transformed into a binary confusion matrix, where the positive class is the class of interest k, and the negative class includes all other classes. The used metric is $MCC_k \in [-1, 1]$, for the binary classification case [135, 136], defined for class k as follows:

$$MCC_k = \frac{\mathrm{TP}\cdot\mathrm{TN} - \mathrm{FP}\cdot\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}} \in [-1, 1]. \tag{3.21}$$

For the multi-class case [137, 138], with K classes, MCC is defined as:

$$\mathrm{MCC} = \frac{c \cdot s - \sum_{k=1}^{K} p_k \cdot t_k}{\sqrt{\left(s^2 - \sum_{k=1}^{K} p_k^2\right)\left(s^2 - \sum_{k=1}^{K} t_k^2\right)}} \in [-1, 1] \tag{3.22}$$

where: $c = \sum_{k=1}^{K} \mathbf{C}_{k,k}$ = total number of correctly classified samples
$s = \sum_{i=1}^{K} \sum_{j=1}^{K} \mathbf{C}_{i,j}$ = total number of samples
$t_k = \sum_{i=1}^{K} \mathbf{C}_{i,k}$ = number of times class k truly occurred
$p_k = \sum_{i=1}^{K} \mathbf{C}_{k,i}$ = number of times class k was predicted

For both cases, the results of MCC are presented in their normalized version, nMCC, linearly projected to the interval $[0, 1]$ as the majority of metrics, defined as:

$$\mathrm{nMCC} = \frac{\mathrm{MCC} + 1}{2} \in [0, 1] \tag{3.23}$$

In addition, Precision (P) and Recall (R) metrics, denoted as P_k and R_k for class k respectively, are also included. In remote sensing, P and R are also known as User's and Producer's accuracy, respectively, defined as:

$$P_k = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \in [0, 1] \tag{3.24}$$

$$R_k = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \in [0, 1] \tag{3.25}$$

The widespread use of Overall Accuracy (OA) is due to its intuitive interpretation of the proportion of correctly classified pixels relative to the total; however, it can conceal biases when the dataset is imbalanced [119]. OA for class k is defined as:

$$\mathrm{OA}_k = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \in [0, 1] \tag{3.26}$$

Each metric provides a different interpretation of the results, and depending on the application, users may prioritize one over the others, although MCC is less biased by the class imbalance of the testing datasets [119]. For instance, when analyzing cloud-free pixels with sufficient data, users tend to prioritize the overestimation of clouds to ensure that the identified pixels are indeed clear skies. In such cases, R is the preferred metric, as it highlights the presence of cloud pixels that have been misclassified as clear sky. In this way, classification reports are provided to end users to help interpret the bias and limitations of classification products.

3.8.2 Testing Classification Report

Given the need to deeply analyze the behavior of the classifier and assess its performance compared to the original PACO masks, a testing report was developed to provide an overview of the results obtained during model testing on independent datasets. It summarizes the previously reviewed classification metrics in a condensed format and is automatically generated after the prediction. In addition, a brief subjective interpretation was conducted for scenes where it was previously known that predictions tended to fail and negatively impact the reported metrics.

Therefore, the implemented methodology is described in this section, allowing for the evaluation of the results, the tools that compose it, and some lessons learned.

3.8.2.1 Metrics Report

After the prediction of the testing dataset, a metrics report was included as a product of this phase. Thanks to the pixel-by-pixel classification approach of the proposed model (instead of tile-by-tile), the prediction can be performed only over the labeled pixels, avoiding the need to classify all pixels from all scenes (if specified) to reduce computation time. In Table 3.11, an example of a testing report is shown, including the confusion matrix and associated classification metrics, as well as a comparison with the baseline model (PACO in this study) to highlight performance differences.

		Ground Truth										
		Clear	**Cloud**	**Shadows**	**Water**	**Snow**	**SUM**	**Precision**	**Recall**	**nMCC**	**Accuracy**	0.69
Prediction	**Clear**	578768	9812	5107	454	783	594924	0.97	0.66	0.71		
	Cloud	46705	46202	752	7	1150	94816	0.49	0.77	0.79	**nMCC**	0.74
	Shadows	36437	1137	64288	4460	9	106331	0.60	0.88	0.86		
	Water	1347	188	648	21317	41	23541	0.91	0.81	0.93		
	Snow	208912	2464	2166	62	6798	220402	0.03	0.77	0.57		
	SUM	872169	59803	72961	26300	8781						
		Clear	**Cloud**	**Shadows**	**Water**	**Snow**	**SUM**	**Precision**	**Recall**	**nMCC**	**Accuracy**	0.60
Baseline	**Clear**	500584	9014	18910	6824	2167	537499	0.93	0.57	0.63		
	Cloud	73808	45349	2627	62	219	122065	0.37	0.76	0.75	**nMCC**	0.67
	Shadows	93861	165	49080	1112	0	144218	0.34	0.67	0.71		
	Water	1087	151	344	18121	107	19810	0.91	0.69	0.90		
	Snow	202829	5124	2000	181	6288	216422	0.03	0.72	0.56		
	SUM	872169	59803	72961	26300	8781						

Table 3.11: Example of a testing report generated after prediction. Comparison between the baseline model results and the predictions from the fitted model.

Different insights can be drawn from Table 3.11. For instance, the total sample counts reveal the strong imbalance in the testing dataset. To support interpretation, all reported values are color-coded: pixel counts range from white-blue (lowest) to dark blue (highest), and performance metrics use a red-to-beige scale (with red indicating the lowest values). The confusion matrix, with its blue shading, shows that most predictions lie along the diagonal, indicating correct classifications. However, a considerable number of misclassifications remain. For example, many *Clear* pixels are misclassified as *Snow*. Conversely, the model tends to confuse *Snow* pixels with *Cloud*, though it improves over previous results by reducing the misclassification of *Shadow* pixels as *Clear*. Regarding the metrics, red shading highlights classes with comparatively lower scores, as in the case of precision and nMCC for *Snow*.

Another example that illustrates how the overall evaluation can be quickly interpreted is shown on the right side of Table 3.11. It is easy to observe the overall increase in nMCC from 0.67 to 0.74, indicating a substantial improvement between the baseline model and the fitted model. Although this improvement is not observed across all the labels. For example, very low precision is observed for the *Snow* class in both cases; however, recall improves from 0.72 to 0.77. Precision and recall for all classes are overall improved, which also results in better nMCC values for each class.

3.8.2.2 Subjective Visual Interpretation

The objective of a classifier is usually to maximize the metrics during the testing phase, and these depend on the reference labels, as the requirements are defined by the labels themselves. However, there is a drawback, as it is difficult to define the completeness of the testing dataset; this task could become a never-ending problem.

To illustrate this situation, a case is briefly discussed during the testing phase of this thesis. A significant improvement was observed by analyzing the obtained classification map and the RGB quick look, shown in Figure 3.17, where a significant number of water pixels are misclassified as clear pixels. However, this is not reflected in the metrics, because most of the water pixels in that area were unlabeled. Thus, even if the improvement is evident, there is no formal scientific way to report it, apart from explaining and displaying the result. This case is further described in the Results chapter 4.

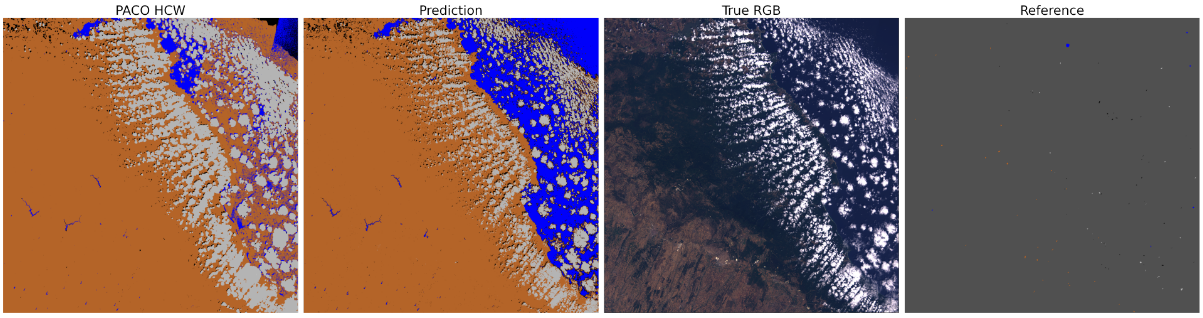


Figure 3.17: Improvement in the predicted water pixels compared to the baseline PACO classification map. Image: Sentinel-2 T35TNG 2023/08/05.

This experience highlighted the need to develop a robust testing methodology, considering various sources to evaluate the results of the fitted models, and even if one of the testing requirements is completely satisfied, it should provide different points of view in an attempt to avoid biases in evaluation. Although subjective interpretation of the results is not well appreciated during scientific exercises, it provides an intuition of the behavior and overall performance, although this insight cannot be used to refine the model.

On the other hand, the same subjective interpretation is widely implemented during the construction of validation and testing exercises, when provided by an independent source, defining the classes based on the interpretation of the labeler, and not on fixed requirements. This was the main motivation to develop the testing methodology presented for the hyperspectral case in subsection 3.3.2, by comparing physical parameters, which are numerically defined and do not change between exercises, allowing for harmonized classification targets.

3.8.3 Regression Metrics

Although this thesis addresses a classification problem, the evaluation of the hyperspectral case is framed as a regression problem. This is because the cloud classification output is transformed into CF based on a patch of EnMAP pixels corresponding to the location of each TROPOMI pixel. The evaluation is conducted by comparing CF values through regression plots, also considering the distribution of samples across different CF ranges through the use of various types of histograms to provide further insight into the statistical significance of the evaluation, along with single-score metrics to compare overall performance. Additional details about the regression plots and histograms are provided in Appendix B.1.

Given two variables with an assumed linear relationship, the following is defined:

$$\Delta_{y_n} = \hat{y}_n - y_n \quad \hat{y}, y \in \mathbb{R} \tag{3.27}$$

where: y = Reference variable,
$\hat{y}$ = Predicted variable,
Δ_{y_n} = Difference between the predicted and reference variables,
n = As a subscript, is the n^{th} sample of N samples.

These metrics are used to evaluate the overall performance of the prediction compared to the reference variable. Each of the metrics provide an assessment of different aspects of the overall fit.

The Root Mean Squared Error (RMSE) measures the accuracy of the prediction and is defined by the square root of the mean of the squared differences, as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\Delta_{y_n}\right)^2} \in [0, \infty) \tag{3.28}$$

The Mean Bias Error (MBE) indicates the average bias based on the sum of the differences, preserving the sign. A positive MBE indicates that the predicted variable overestimates the reference values, while a negative value indicates underestimation.

$$\mathrm{MBE} = \frac{1}{N}\sum_{n=1}^{N}\Delta_{y_n} \in (-\infty, \infty) \tag{3.29}$$

Lastly, the coefficient of determination, denoted as R^2, indicates the model fit. A value of $R^2 = 1$ corresponds to a perfect fit, while values of $R^2 \leq 0$ indicate that the prediction performs worse than a horizontal line predicting the mean $\bar{y}$.

$$R^2 = 1 - \frac{\sum_{n=1}^{N}\left(\Delta_{y_n}\right)^2}{\sum_{n=1}^{N}\left(y_n - \bar{y}\right)^2} \in (-\infty, 1] \tag{3.30}$$

where: $\bar{y}$ = Mean of the reference variable y over N samples.

Although these are informative single-score metrics, the analysis should always be complemented with regression plots and histograms to support the statistical evaluation and provide a deeper understanding of the regression results.

3.8.3.1 APU Plots

For statistical analysis of the regression, the use of Accuracy, Precision and Uncertainty (APU) plots [7, 139, 140, 141] enables evaluation across different ranges of the reference variable. APU plots are commonly used to compare various physical parameters in the ESA Atmospheric Correction Inter-comparison eXercise (ACIX) exercises [142, 143].

The APU plot begins with the discretization of the range of the reference variable by creating a histogram. Then, three evaluation metrics are computed for each bin of the reference variable, revealing different aspects of the regression performance, while taking into account the statistical significance based on the number of samples in each bin. Therefore, the difference $\Delta_{y_n} = \hat{y}_n - y_n$ is also used in the APU plots. However, in contrast to the previous definition in equation (3.27), it is now evaluated bin by bin from the histogram, where the subscript n represents the n^{th} sample of N_b samples in bin b out of B total bins.

The Accuracy (A) reflects the mean bias, the Precision (P) represents the repeatability of the predicted variable by computing the standard deviation with the mean bias corrected, and the Uncertainty (U) is the statistical deviation of the predicted variable from the reference [139]. The APU metrics are then defined in equations (3.31), (3.32), and (3.33).

$$A = \frac{1}{N_b} \sum_{n=1}^{N_b} \Delta_{y_n} \tag{3.31}$$

$$P = \sqrt{\frac{1}{N_b - 1} \sum_{n=1}^{N_b} (\Delta_{y_n} - A)^2} \tag{3.32}$$

$$U = \sqrt{\frac{1}{N_b} \sum_{n=1}^{N_b} (\Delta_{y_n})^2} \tag{3.33}$$

Note that A (equation 3.31) is equivalent to the MBE (equation 3.29), and similarly, U (equation 3.33) corresponds to the RMSE (equation 3.28). Although mathematically equivalent, these metrics differ conceptually. In the APU plot, the metrics are computed per bin and reflect the statistical significance of each bin, depending on its sample population. In contrast, the metrics defined in subsection 3.8.3 are computed over the entire dataset, providing a global assessment.

In Figure 3.18, an example of an APU plot is shown, displaying the three described metrics across all bins spanned by the reference variable. The plot examples in this subsection are generated using $N = 1500$ synthetic samples from a uniform distribution with additive normal noise, such that $\{y \sim U(a = 0, b = 100)\}$ and $\{\hat{y} = y + \mathcal{N}(\mu = 0, \sigma = 5) \,|\, \hat{y} = \min(\max(\hat{y}, 0), 100)\}$.

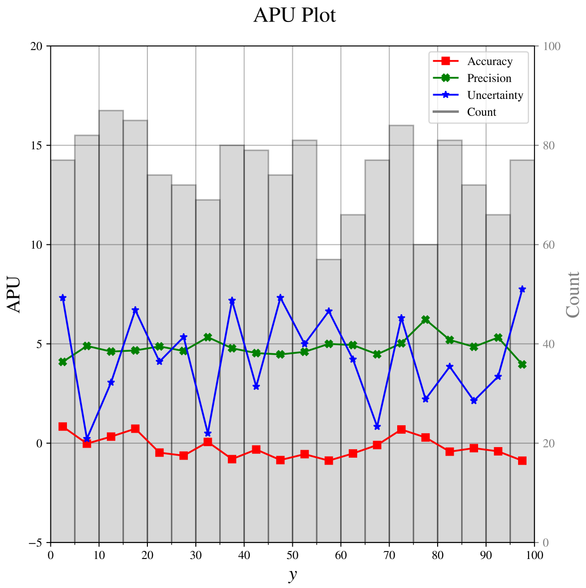


Figure 3.18: Example of an APU plot showing Accuracy (A), Precision (P), and Uncertainty (U) metrics across $B = 20$ bins using $N = 1500$ samples.

Chapter 4

Results and Discussion

This chapter presents the results obtained in this thesis, divided into two main sections, as described in the case studies presented in Section 3.3:

- The multispectral case using Sentinel-2 imagery, evaluating the classification with independently labeled datasets.
- The hyperspectral case using EnMAP imagery, translating the evaluation from a classification product into physical parameters by comparing the obtained cloud maps with the TROPOMI CF products from the Sentinel-5P mission.

Therefore, the following sections discuss the results using the workflow and tools presented in Chapter 3, providing insights, hypotheses, and analyzing the behavior of the fitted models with different training sets.

4.1 Results for the Multispectral Case

This section presents the evaluation of the predictions obtained from the trained models compared with the base pre-classification maps from PACO, hereafter PACO "*hcw*" (see [12]). Table 4.1 provides an overview of the results obtained for each testing dataset (in terms of $nMCC$, darker blue = higher), produced by the three tested models, CNN1D, 2D and 3D, for the three pixel-selection strategies—*no filter*, *physics filter*, and *uniclass filter* (Section 3.5)—. The metrics are then compared in terms of absolute improvement, expressed as Percentage Points (pp). All values are single-run scores; variance analysis is left for future work due to the high computing cost.

Table 4.1 summarizes the normalized Matthews correlation coefficient ($nMCC$) for all models and pixel-selection strategies. Across both test datasets, every CNN variant outperforms the PACO's "*hcw*" baseline by 1.8–18.3pp.

- On TD1, the best result is 0.888 $nMCC$ obtained by CNN2D-*uniclass filter*, a gain of +3.3pp over PACO "*hcw*".
- On TD2, the largest improvement comes from CNN1D-*uniclass filter* (0.848 $nMCC$, +18.3pp), followed closely by CNN3D-*uniclass filter* (0.833 $nMCC$, +16.8pp)

Model		**TD1**	**TD2**
PACO		0.855	0.665
CNN1D	*No filter*	0.867	0.723
	Physics filter	0.877	0.848
	Uniclass filter	0.879	0.764
CNN2D	*No filter*	0.884	0.730
	Physics filter	0.883	0.755
	Uniclass filter	0.888	0.732
CNN3D	*No filter*	0.872	0.734
	Physics filter	0.879	0.741
	Uniclass filter	0.874	0.833

Table 4.1: Overview of the obtained classification results across all testing datasets. The results are expressed in $nMCC$; darker blue indicates higher scores per testing dataset.

These gains are observed across the diverse surfaces and atmospheric conditions represented in TD1/TD2. We attribute them to the richer feature extraction of the CNNs and to the integration of inter-class relationships within a single network.

TD2 includes many complex or ambiguous pixels, hence baseline scores are lower than for TD1. This gap is narrowed by CNN1D-*physics filter* and CNN3D-*uniclass filter*. These results suggest that 2D spatial filtering best suits the average cases of TD1 set, whereas spectral-focused 1D kernels cope better the ambiguous pixels in TD2. CNN3D offers a strong but more expensive compromise ($\approx$ 305 k vs. 77 k parameters).

Pixel-selection strategies improve over the *no filter* baseline. On TD2, the *physics filter* lifts CNN1D by +12.5pp, while the *uniclass filter* adds +9.9pp to CNN3D. On TD1 the gains are smaller ($\leq$ 1.2pp), and CNN2D-*physics filter* shows a marginal drop of -0.1pp.

The larger gains on TD2 reflect how each filter targets the dominant source of label noise in that set. The *physics* and *uniclass filters* remove pixels whose spectral indices fall near class-decision thresholds, eliminating many mixed or ambiguous spectra; this particularly benefits the spectral feature extraction of CNN1D and CNN3D, which rely on the spectral signature to resolve uncertainty, but also benefits CNN2D in a more limited way. In TD1, the original labels are already cleaner and more balanced, hence both filters bring only marginal additional benefit (or a slight loss when informative edge pixels are discarded, as with CNN2D-*physics filter*).

4.1.1 Per-class Results Analysis

In order to achieve a deeper understanding of the behavior of the proposed pixel selection strategies, the results for each class are presented individually in Tables 4.2 and 4.3, corresponding to TD1 and TD2, respectively. For ease of reading, only significant improvements or losses are discussed in the text.

For both testing datasets, the biggest improvement is observed for the *shadow* class, where $nMCC$ is considerably improved by 4–9pp for TD1 and 16.7–20.2pp for TD2, consistently across both datasets.

The next best improvement is observed for the *clear* class, improved by 1.1–4.1pp for TD1 and 7.2–22.7pp for TD2, highlighting CNN1D-*physics filter* and CNN3D-*uniclass filter* for TD2, which achieve the best results by a considerable margin. This improvement is correlated with the improvement of the *snow* class in TD2 for the aforementioned CNNs, which is not improved remarkably for any other CNN, and even loses performance in some cases.

The *atmosphere* class is generally improved, except for CNN1D-*no filter* for TD2, with improvements of $\leq$ 3pp for TD1 and $\leq$ 6.1pp for TD2. On the contrary, the *water* class shows improvements of $\leq$ 1.3pp for TD1, but most of the trained models show losses of $\leq$ 2.4pp for TD2, except for CNN1D-*no filter* and CNN2D-*uniclass filter*, which show an improvement of $\leq$ 1pp.

To pinpoint where the remaining errors occur and which classes are most affected, we compare the confusion matrices of the best CNN for each test set with the PACO

		TD1				
	Model	*Clear*	Atmos	Shadow	Water	*Snow*
	PACO	0.861	0.855	0.769	0.905	0.906
CNN1D	*No filter*	0.874	0.873	0.809	0.911	0.822
	Physics filter	0.876	0.878	0.834	0.915	0.889
	Uniclass filter	0.872	0.885	0.834	0.916	0.905
CNN2D	*No filter*	0.899	0.885	0.844	0.910	0.821
	Physics filter	0.898	0.878	0.850	0.915	0.832
	Uniclass filter	0.902	0.881	0.859	0.918	0.861
CNN3D	*No filter*	0.878	0.873	0.832	0.911	0.825
	Physics filter	0.884	0.877	0.842	0.912	0.878
	Uniclass filter	0.872	0.866	0.845	0.922	0.867

Table 4.2: Obtained classification results for TD1 presented for each class separately. The results are expressed in $nMCC$; darker blue indicates higher scores per class.

		TD2				
	Model	*Clear*	Atmos	Shadow	Water	*Snow*
	PACO	0.630	0.746	0.712	0.895	0.558
CNN1D	*No filter*	0.702	0.710	0.914	0.905	0.543
	Physics filter	0.857	0.807	0.900	0.896	0.655
	Uniclass filter	0.757	0.794	0.891	0.883	0.577
CNN2D	*No filter*	0.712	0.749	0.880	0.891	0.556
	Physics filter	0.748	0.758	0.881	0.882	0.561
	Uniclass filter	0.716	0.756	0.879	0.901	0.564
CNN3D	*No filter*	0.715	0.756	0.898	0.894	0.551
	Physics filter	0.730	0.761	0.885	0.871	0.565
	Uniclass filter	0.843	0.772	0.893	0.877	0.726

Table 4.3: Obtained classification results for TD2 presented for each class separately. The results are expressed in $nMCC$; darker blue indicates higher scores per class.

hcw baseline. For TD1 the top performer is CNN2D-*uniclass filter*, whereas for TD2 it is CNN1D-*physics filter*. Tables 4.4 and 4.5 show, for each model, the percentage of pixels predicted in every class (rows) versus the reference labels (columns). The last two columns report class-wise precision (P) and recall (R), allowing over- and under-estimation to be read directly.

Table 4.4: Confusion matrices for TD1. Top: CNN2D-uniclass filter; bottom: PACO hcw baseline. Each cell is column percentage; darker blue = higher.

		TD1 - Reference labels						
Model	**Class**	*Clear*	*Atmos*	*Shadows*	*Water*	*Snow*	P	R
CNN2D	*Clear*	89.1	10.0	10.9	2.4	0.0	0.86	0.89
	Atmos	7.6	85.0	11.1	6.8	10.0	0.83	0.85
	Shadows	1.8	0.2	71.0	11.8	0.0	0.81	0.71
	Water	0.8	0.0	3.7	79.0	0.0	0.93	0.79
	Snow	1.0	4.7	3.4	0.1	89.9	0.60	0.90
PACO	*Clear*	91.4	16.2	30.0	7.6	0.2	0.75	0.91
	Atmos	6.0	78.3	16.5	6.6	0.4	0.82	0.78
	Shadows	2.0	0.1	43.9	7.2	0.0	0.78	0.44
	Water	0.5	0.0	8.8	78.6	0.0	0.89	0.79
	Snow	0.1	5.3	0.8	0.0	99.4	0.68	0.99
	Total	637,991	577,872	267,190	274,884	71,124		

According to the confusion matrix in Table 4.4, the CNN2D model reduces a considerable number of errors that the PACO *hcw* baseline makes between *shadow* pixels classified as *clear*: *shadow* recall rises from 0.44 to 0.71, and precision for the *clear* class increases from 0.75 to 0.86. The cost of this gain is an increase in the number of *snow* pixels misclassified as *atmosphere*, with recall dropping from 0.99 to 0.90 and precision from 0.68 to 0.60.

Table 4.5: Confusion matrices for TD2. Top: CNN1D-physics filter; bottom: PACO hcw baseline. Formatting as in Table 4.4.

		TD2 - Reference labels						
Model	**Class**	*Clear*	*Atmos*	*Shadows*	*Water*	*Snow*	P	R
CNN1D	*Clear*	92.2	20.3	10.3	5.6	25.7	0.97	0.92
	Atmos	4.6	77.9	0.6	0.0	20.8	0.53	0.78
	Shadows	1.3	1.1	85.4	22.6	0.1	0.78	0.85
	Water	0.1	0.3	0.9	71.0	1.7	0.89	0.71
	Snow	1.8	0.5	2.8	0.8	51.7	0.20	0.52
PACO	*Clear*	57.4	15.1	25.9	25.9	24.7	0.93	0.57
	Atmos	8.5	75.8	3.6	0.2	2.5	0.37	0.76
	Shadows	10.8	0.3	67.3	4.2	0.0	0.34	0.67
	Water	0.1	0.3	0.5	68.9	1.2	0.91	0.69
	Snow	23.3	8.6	2.7	0.7	71.6	0.03	0.72
	Total	872,169	59,803	72,961	26,300	8,781		

In Table 4.5, the CNN1D model largely removes misclassification in the *clear* class, improving recall from 0.57 to 0.92. On the other hand, *shadow* precision goes up from 0.34 to 0.78, with recall rising from 0.67 to 0.85. The main remaining weakness is that some *snow* pixels are labeled as *atmosphere*, so snow recall drops from 0.72 to 0.52, although their precision increases from 0.03 to 0.20.

4.1.2 Qualitative Analysis

Because the testing datasets contain only sparse reference (or ground-truth) labels, this subsection complements the quantitative scores with a qualitative comparison of classification maps for representative scenes.

The observed improvement of the *snow* class for TD2 is shown in Fig. 4.1, where a dried salty lake is misclassified as snow and cloud by PACO *hcw*. In addition, most of the *clear* pixels of TD2 are labeled in this scene, causing the observed low $nMCC$ for the *snow* class. The reason this problem frequently occurs is likely due to high-reflectance bright soil being often misclassified as snow or clouds. This issue is almost solved by CNN1D-*physics filter* and CNN3D-*uniclass filter*, where the majority of pixels are correctly classified in the *clear* class, while CNN2D-*uniclass filter* continues misclassifying these pixels. On the other hand, all models correct the misclassification of shadows caused by low-reflectance soil (black soil).

Fig. 4.2 shows how the three pixel-selection strategies affect CNN2D. While Table 4.1 suggests only a 0.5 pp spread between the CNN2D variants, the qualitative map reveals

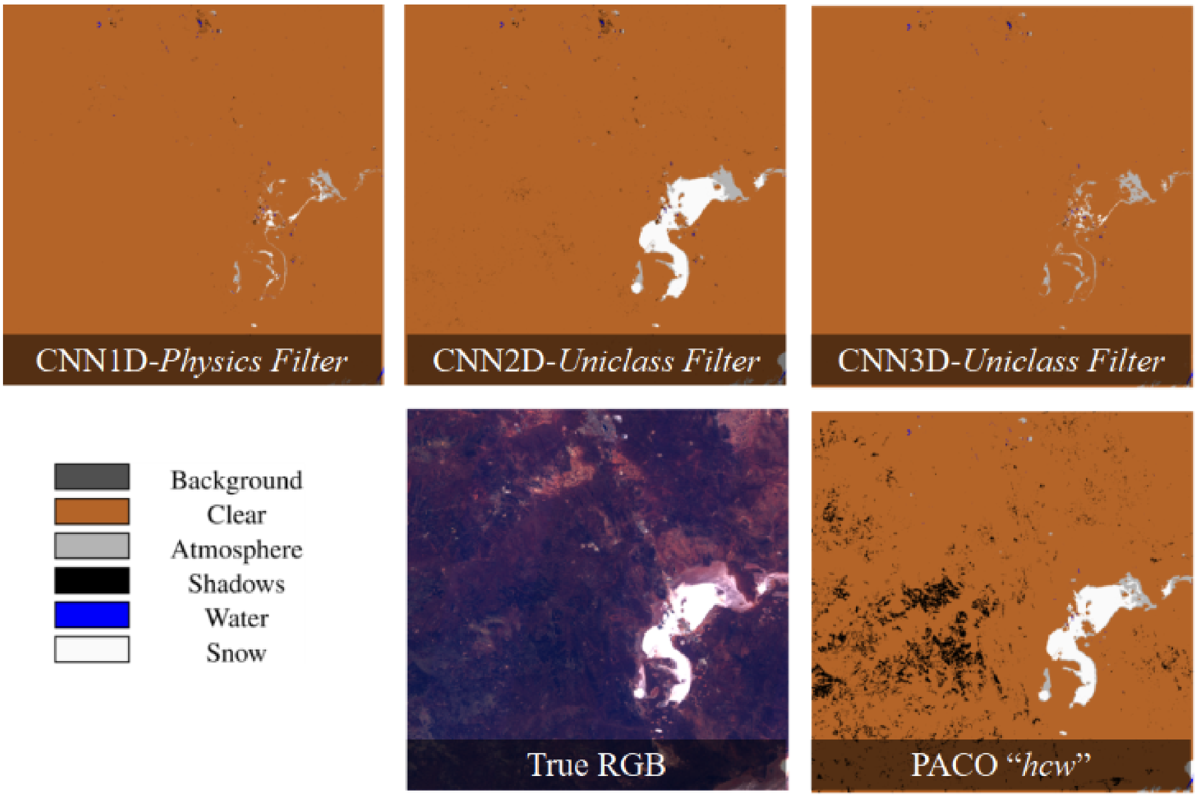


Figure 4.1: Qualitative comparison between PACO *hcw* and the prediction from tested models with true RGB image for reference. Image: Lake Lefroy, Australia. Sentinel-2, TD2 T51JUF, 2018/08/19.

larger practical differences. Only the *uniclass filter* variant succeeds in labeling the narrow cloud-shadow pixels that lie on top of the highly reflective snow. By contrast, the *physics* and *no filter* models, like the PACO hcw baseline, misclassify these shadow pixels as water.

A persistent compromise of all three CNN variants is their systematic confusion of *snow* with *cloud*. In Table 4.2 the *snow* class is the only one that consistently loses accuracy. Fig. 4.2 makes this clear: regions that should be labeled *snow* (and in a few cases even *clear*) are frequently reclassified as *cloud* by the CNNs, whereas the PACO *hcw* baseline keeps them as *snow*. The error is not surprising: fresh *snow* and optically thick *cloud* have similar high-reflectance spectra, making them hard to separate.

Conversely, Fig. 4.3 highlights the opposite side of the trade-off. Here the PACO *hcw* baseline mistakes a bank of optically thick *clouds* for *snow*, whereas all three CNN models correctly assign the *cloud* label.

A case that we would like to highlight is shown in Fig. 4.4, where a significant amount of *water* pixels are misclassified in the *clear* class. This error is due to a well-known

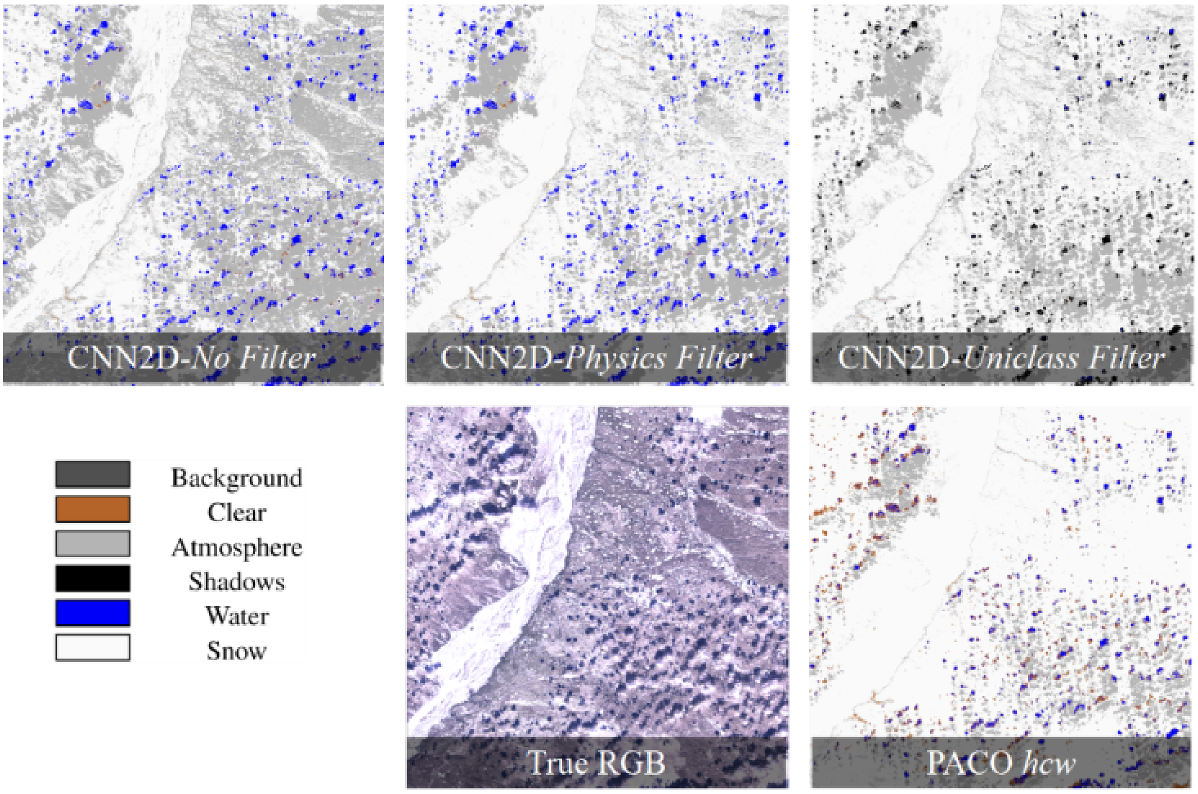


Figure 4.2: Qualitative comparison between PACO *hcw* and the CNN2D prediction with true RGB image for reference. Image: Yakutsk, Russia. Sentinel-2, TD1 T52VEP, 2021/04/16.

parallax effect that causes a stripping signature in Sentinel-2 imagery [144, 145] caused by different viewing angles of onboard sensors. This error is not reflected in the testing dataset given that almost no pixels are labeled in this area, but the misclassification is clearly visible. Our approaches solved this problem, which is attributed to the improvement in the generalization of spectral signature recognition.

The gains in performance for all trained models are nevertheless consistent, and most evident for the models trained with the *physics* and *uniclass* pixel-selection strategies. Outside the presented scenes, most classification maps look visually similar to the PACO *hcw* baseline, with only modest refinements.

Figure 4.3: Qualitative comparison between PACO *hcw* and the CNN3D-*uniclass filter* prediction with true RGB image for reference. Because the remaining models perform similarly, their classification maps are omitted to avoid redundancy. Image: Indonesia. Sentinel-2, TD1 T48MZT, 2021/11/22.

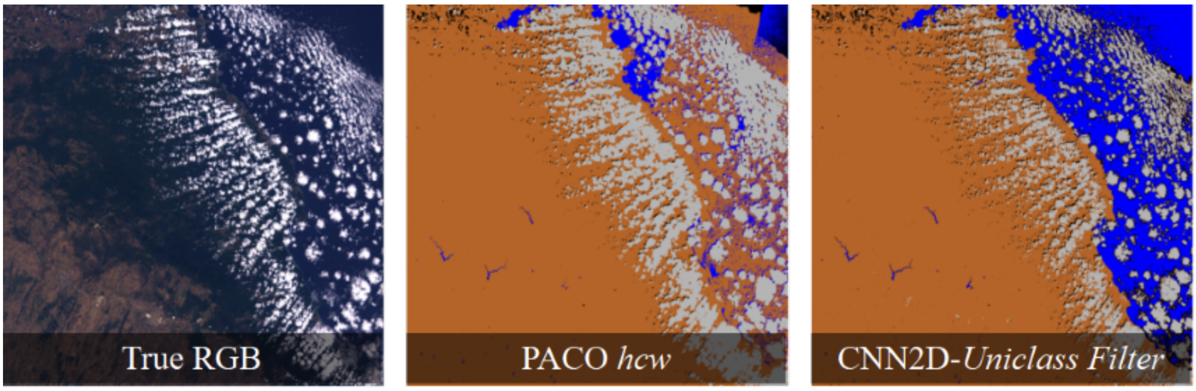


Figure 4.4: Qualitative comparison between PACO *hcw* and the CNN2D-*uniclass filter* prediction with true RGB image for reference. Because the remaining models perform similarly, their classification maps are omitted to avoid redundancy. Image: Bulgaria/Turkey. Sentinel-2, TD1 T35TNG, 2023/08/05.

4.2 Results for the Hyperspectral Case

This section presents the evaluation of the model's predictions compared with the cloud products from the Sentinel-5P mission. As mentioned in Section 3, the classification maps from PACO masking and those produced by the CNN2D model for each training dataset (obtained using the *no filter*, *physics filter*, and *uniclass filter* selection strategies) are transformed into CF values corresponding to the pixel locations of Sentinel-5P. The metrics and plots used for the evaluation of the results are described in Subsection 3.8.3.

To better interpret the results comparison, the following subsections first examine some aspects that arose during the evaluation, related to the distribution of the pixels and the main sources of uncertainty for CF comparison. These include the time difference between the captures from EnMAP and Sentinel-5P, as well as the differences between the Radiometric CF from Sentinel-5P and the Geometric CF from EnMAP.

Therefore, it is defined for ease of reading:

$$\hat{y} = \text{Predicted CF using EnMAP imagery,} \tag{4.1}$$

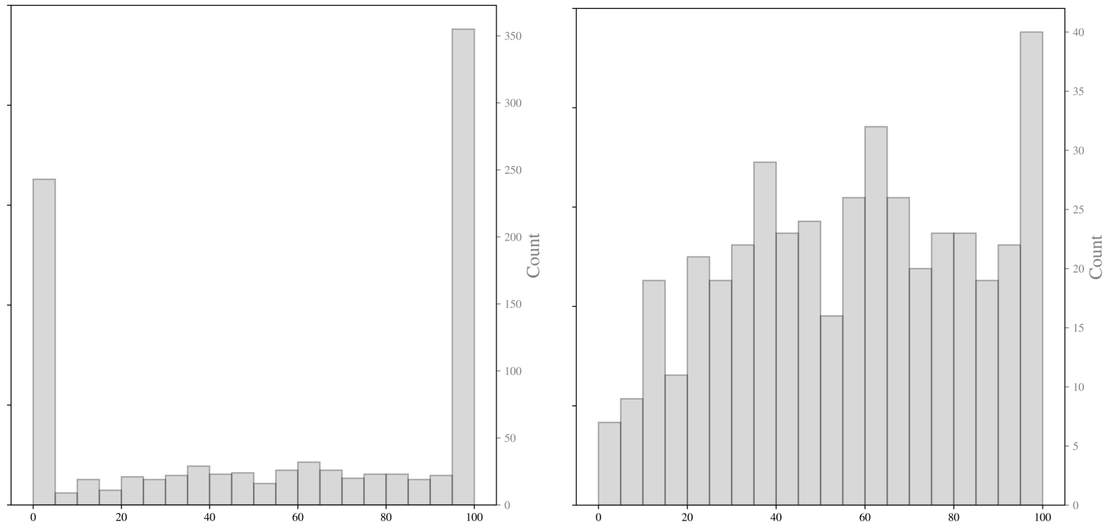
$$y = \text{Reference CF.} \tag{4.2}$$

4.2.1 Impact of Cloud-Free or Cloud-Full Coverage

Depending on the location and season, cloud coverage and its distribution vary. As seen in Subsection 3.4.4, most of the matching overpasses between EnMAP and Sentinel-5P acquired within a short time window were captured either under cloud-free conditions or, conversely, completely covered by clouds. A histogram of the CF distribution, pixel by pixel, is shown in Figure 4.5, where it is clearly visible that most of the available pixels for training are concentrated at CF = 0% (cloud-free) and CF = 100% (cloud-full).

The impact of the CF distribution is evident when examining the correlation metrics. Cloud-free and cloud-full pixels are usually correctly classified (particularly in areas where the surface is not highly reflective, e.g., snow, ice, white sand, or greenhouses), resulting in a difference between prediction and reference CF of $\Delta_{y_n} = 0$. Consequently, when computing correlation metrics (e.g., R^2), the dominance of perfectly matched pixels masks the model's performance on the remaining pixels, obscuring the classifier's behavior across the full CF range. Figure 4.6 shows regression plots comparing the cases where cloud-free and cloud-full pixels are included vs. excluded. The prediction shown ($\hat{y}$) is the baseline CF from PACO.

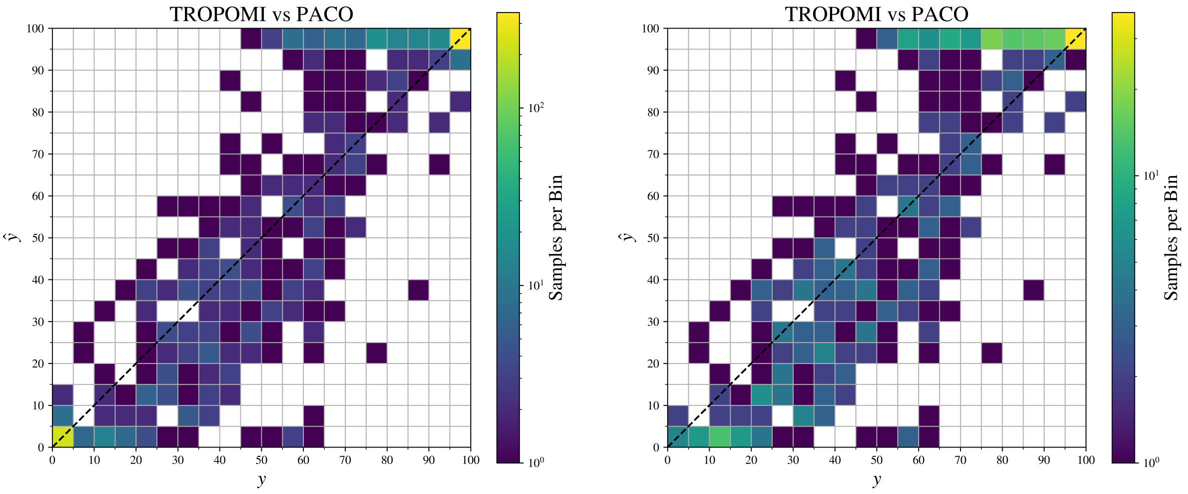
Figure 4.6 shows that, even though the distribution in the regression plot appears similar in both cases, the correlation metric R^2 drops drastically by -0.41 Percentage Points (%pt), highlighting the importance of performing evaluation assessments using a combination of metrics and plots.



(a) Including all pixels.

(b) Excluding cloud-free and cloud-full pixels.

Figure 4.5: CF distribution of Sentinel-5P pixels in testing dataset



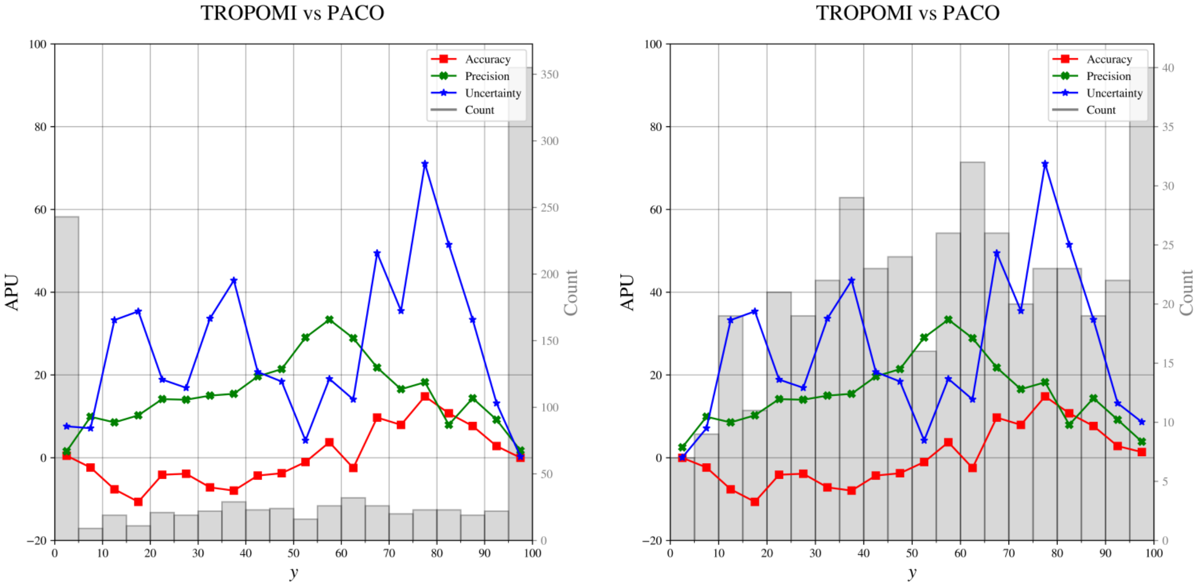
(a) $R^2 = 0.9$. Including all pixels.

(b) $R^2 = 0.49$. Excluding cloud-free and cloud-full pixels.

Figure 4.6: Comparison of 2D histograms including or excluding cloud-free and cloud-full pixels for regression assessment. Note the difference between color bars scale.

These pixels are part of the evaluation and contribute to the statistics, as failing on these “easy cases” must be considered a very harmful behavior. Consequently, misclassifying them would significantly decrease the regression metrics. However, when a large number of these pixels is consistently correctly classified, they can mask the model's performance on the remaining pixels.

To mitigate this effect, APU plots provide an overview for evaluating across the whole CF range, dividing into bins and performing the evaluation for each one. In this way, if a bin is overrepresented, it does not affect the evaluation of the rest. In Figure 4.7, the corresponding APU plots corresponding to the cases presented in 4.6 are shown.



(a) Including all pixels.

(b) Excluding cloud-free and cloud-full pixels.

Figure 4.7: Comparison of APU plots, with and without cloud-free and cloud-full pixels for regression assessment.

Figure 4.7 demonstrates that despite the high amount of pixels in the side-extreme bins of the histogram, the rest of the evaluation on the remaining bins stays unchanged, allowing independent evaluation of all sample types. Therefore, during the evaluation of this specific study, it is recommended to consistently refer to the APU plots. In the rest of the plots, a logarithmic scale allows differentiation of small changes.

4.2.2 Comparison of Radiometric and Geometric CF

This subsection provides an analysis of the comparison of Geometric Cloud Fraction (GCF), obtained by calculating the area covered by clouds using an EnMAP image, and the Radiometric Cloud Fraction (RCF), computed from the COT measured by Sentinel-5P. In the validation exercise of Sentinel-5P cloud products, Compernolle et al. [128] compares the RCF of Sentinel-5P to RCF obtained from Suomi-NPP-VIIRS and MODIS, providing the following statements [128]:

- "For optically thin clouds, the radiometric cloud fraction is smaller than the geometric one"
- "In most cases, RCF $\leq$ GCF"
- "It should be noted that the performance of OCRA/ROCINN is optimal for high geometric cloud fractions and optically thick clouds."

From the presented results of this thesis, this behavior is observed through the APU plots through the evaluation of accuracy, revealing overestimation or underestimation of the CF. For example, in Figure 4.7, a systematic GCF ($\hat{y}$) overestimation is observed at high values compared to RCF (y) (between 65% and 95%), but underestimation of GCF at low values (between 5% and 50%) compared to RCF.

In Figure 4.8, a case is presented where GCF is overestimated compared to the RCF, given that optically thin clouds cause lower RCF.

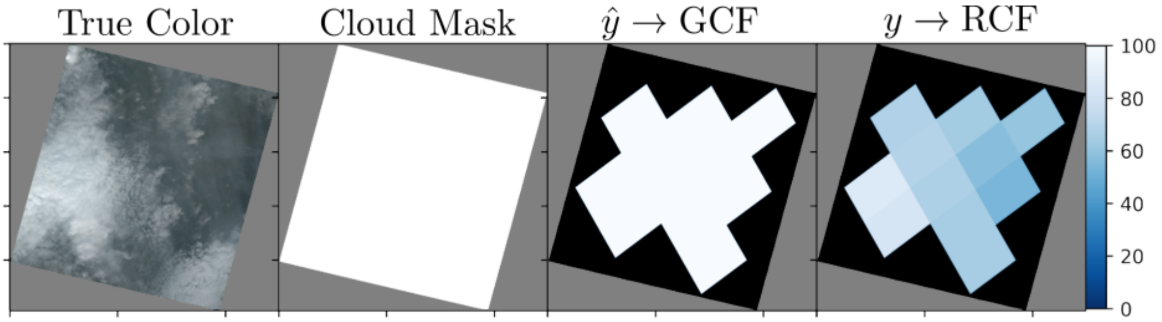


Figure 4.8: Image: EnMAP DT38193, tile-003. Captured at 2023/08/30. Time difference between captures of 25 s.

On the other hand, in Figure 4.9, a case is presented where GCF is underestimated compared to the RCF, as discussed for Figure 4.7 for the low range of CF. RCF appears to be overestimated by the reference, although as mentioned, "In most cases, RCF $\leq$ GCF" in [128], with these results, this reveal a clear trend in the APU plot in Figure 4.7 for low RCF.

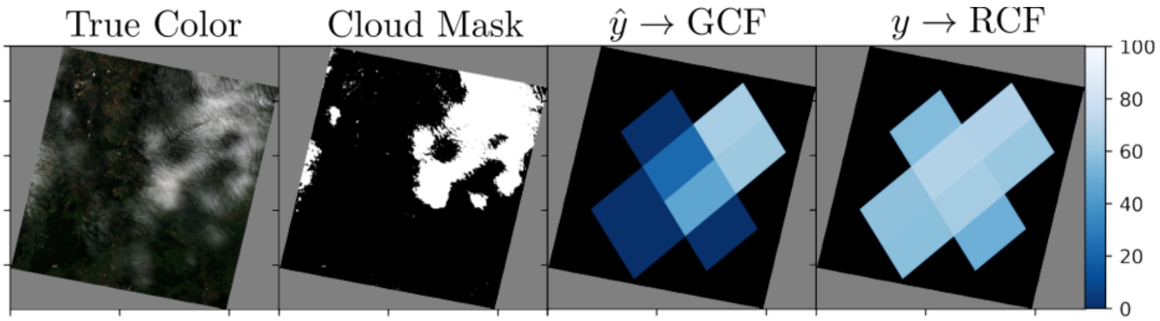


Figure 4.9: Image: EnMAP DT44686, tile-030. Captured at 2023/10/07. Time difference between captures of 46 s.

In addition, in the ATBD of Sentinel-5P cloud products [146], an uncertainty requirement of 20% is reported for RCF. This context makes GCF and RCF not directly comparable [128]. Therefore, the potential uncertainty in the evaluation should be considered together with the bias information provided by APU plots.

4.2.3 Effect of Time Difference Between Captures

As mentioned in Subsection 3.4.4, the time difference between captures of EnMAP and Sentinel-5P is the main source of uncertainty when comparing RCF and GCF.

There is a trade-off between including matching overpasses with low time difference and the completeness of the testing dataset. As both missions continue capturing images, this trade-off vanishes, as more matching overpasses with low time difference between captures become available. In this section, the mentioned trade-off is evaluated, in order to find an acceptable compromise between time difference and completeness.

In Table 4.6, the regression metrics are shown by including in the testing datasets pixels with time differences ranging from 30 s to 180 s.

Time Difference [s]	**Pixels Count**	**R^2**	**MBE**	**RMSE**
30	170	0.85	6.07	4.06
60	325	0.81	4.04	4.15
90	610	0.89	2.66	3.68
120	746	0.91	2.37	3.53
150	842	0.91	1.34	3.59
180	982	0.90	0.30	3.58

Table 4.6: Regression metrics for time differences ranging from 30 s to 180 s, including all pixels.

In the same manner, in Table 4.7, the regression metrics are shown, this time filtering out the cloud-free and cloud-full cases.

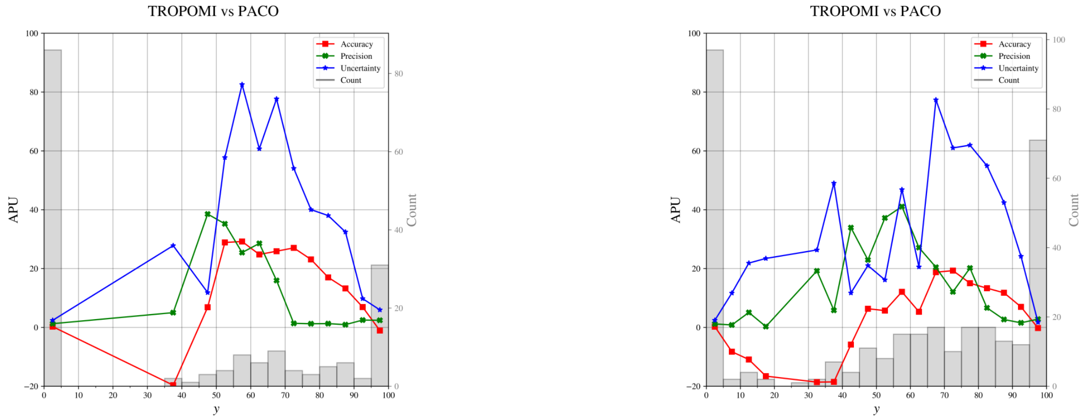
Time Differece [s]	**Pixels Count**	**R^2**	**MBE**	**RMSE**
30	53	-3.43	19.66	5.43
60	171	-0.27	7.74	4.86
90	270	0.44	5.84	4.50
120	295	0.46	5.79	4.45
150	340	0.46	3.14	4.50
180	431	0.49	0.55	4.39

Table 4.7: Regression metrics for time differences ranging from 30 s to 180 s, excluding cloud-free and cloud-full pixels.

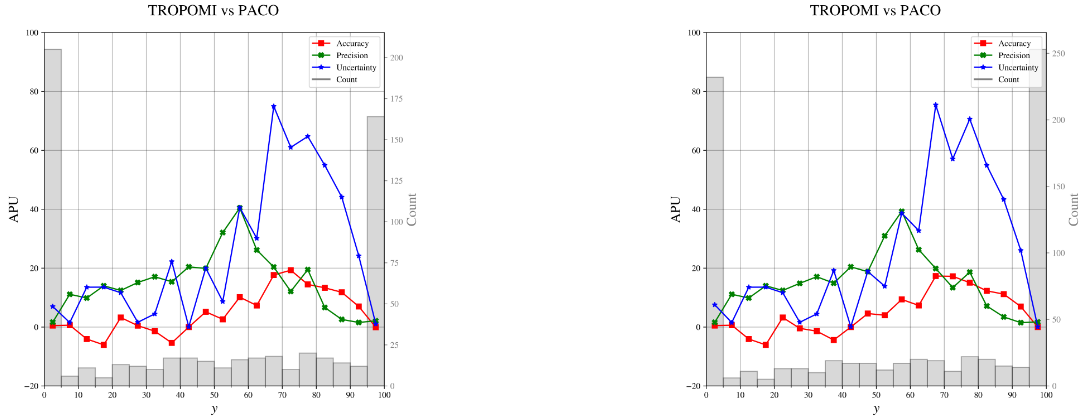
In both cases, completeness on testing datasets seems to favor the regression metrics, as in both cases, the smallest MBE is obtained with a time difference of 180 s. In addition, the RMSE and R^2 present an almost consistent decreasing trend as the pixel count increases. Therefore, completeness of the testing datasets is preferred based on the regression metrics in this testing exercise.

In addition, the APU plots and 2D Histograms for each time difference, including all pixels in the testing dataset, are presented in Figures 4.11 and 4.10, respectively.

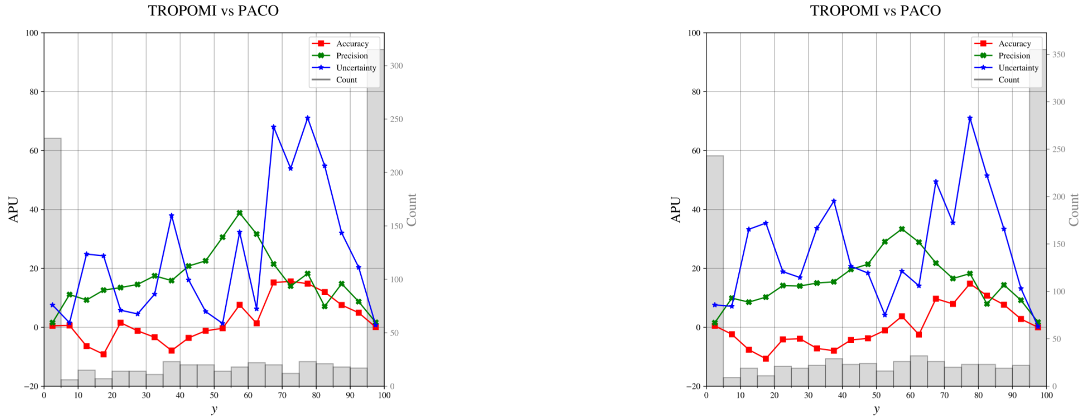
Despite the daily-global coverage of Sentinel-5P, the relatively small count of EnMAP scenes makes the matching overpasses with a time difference under 1 minute not enough to compute metrics with low uncertainty, as observed in Figure 4.10. This is solved by increasing the allowed time difference, at the expense of increasing the uncertainty described in Subsection 3.4.4, mainly caused by the movement of clouds. Therefore, although the correlation is clear, this methodology will be optimal for testing missions with high volumes of data, featuring wide Earth coverage and high revisit frequency.



(a) 30 s (b) 60 s

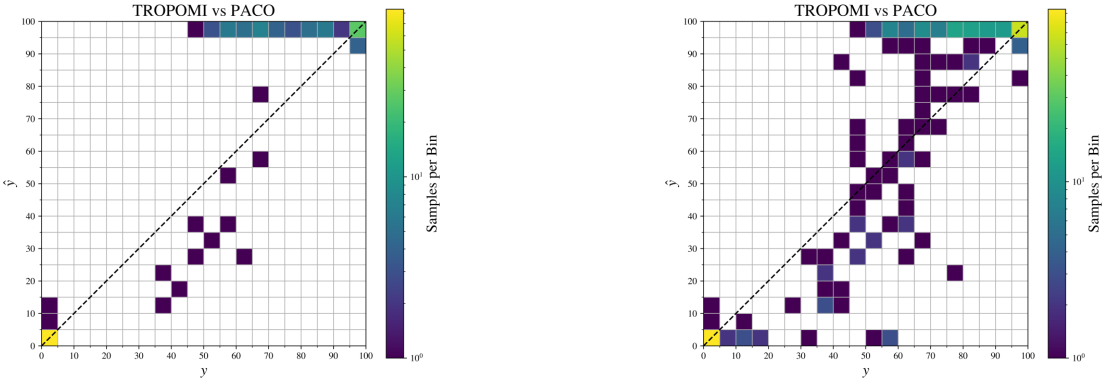


(c) 90 s (d) 120 s

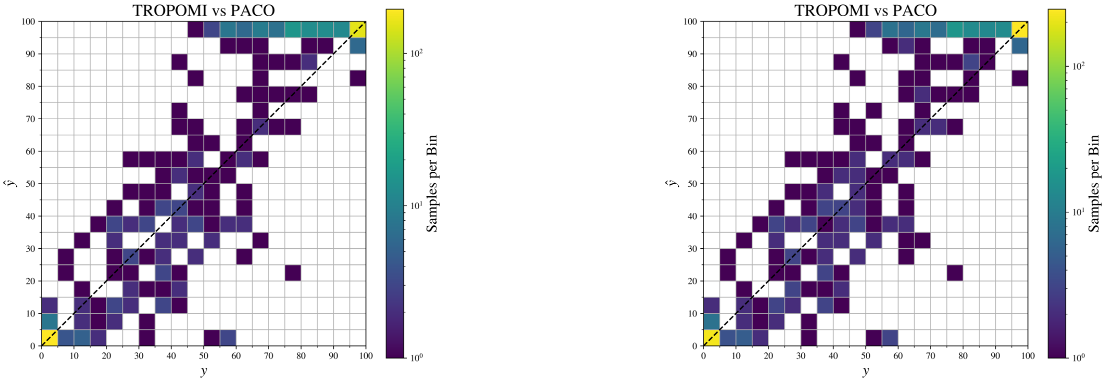


(e) 150 s (f) 180 s

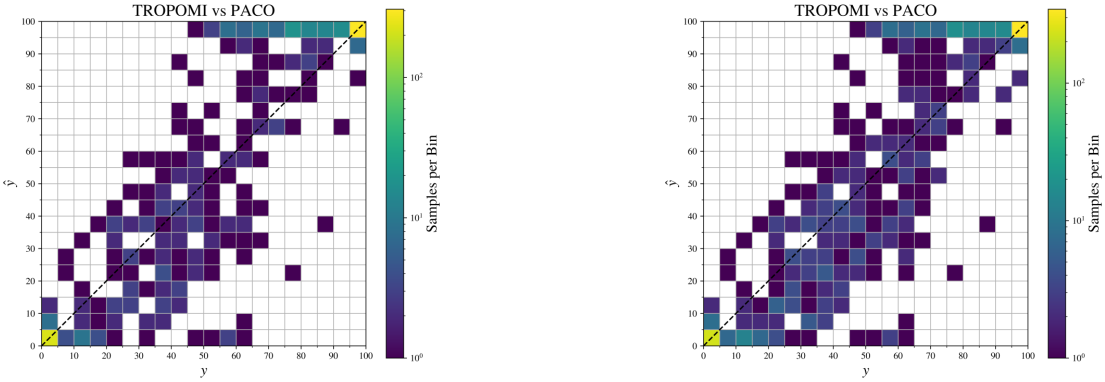
Figure 4.10: APU plots comparing TROPOMI and PACO across different time differences ranging from 30 s to 180 s.



(a) 30 s (b) 60 s



(c) 90 s (d) 120 s



(e) 150 s (f) 180 s

Figure 4.11: 2D histograms comparing TROPOMI and PACO at different time differences, from 30 s to 180 s.

4.2.4 Results Comparison

In this subsection, the comparison is presented of the obtained results by trained CNNs with the baseline pre-classification from PACO, both evaluations taking as a reference de RCF from Sentinel-5P.

As discussed before in this section, the comparison is performed by including all pixels under less than 180 s of time difference between captures, including all the discussed plots for results evaluation. In Table 4.8 the results are shown, reporting the regression metrics (R^2, MBE, and RMSE) for each CNN model trained with different pixel selection strategies, compared with the baseline PACO.

Classifier		**R²**	**RMSE**	**MBE**
PACO		0.90	3.58	0.30
CNN1D	*No-Filter*	0.90	3.63	1.07
	Uniclass	0.89	3.66	0.52
	Physics	0.88	3.75	-0.19
CNN2D	*No-Filter*	0.90	3.60	0.92
	Uniclass	0.90	3.59	1.11
	Physics	0.88	3.75	0.00
CNN3D	*No-Filter*	0.89	3.67	0.34
	Uniclass	0.89	3.69	0.27
	Physics	0.88	3.76	0.18

Table 4.8: Regression metrics. Testing dataset include all pixels under 180 s between captures

.

Although some differences are observed between the CNN models and the PACO baseline, the results remain comparable across all metrics, with R^2 values between 0.88–0.90 and RMSE differences generally below 0.2. These limited variations suggest that the improvements introduced by the pixel selection strategies may not be easily detectable with this testing approach. Given the large GSD of Sentinel-5P relative to EnMAP, finer spatial patterns captured by the CNN models may be averaged out when aggregated to match Sentinel-5P's pixel footprint. As a result, the added value of these strategies, demonstrated in the multispectral case experiments, may be obscured in this particular evaluation and favor the bias of the base PACO baseline classification.

A trade-off is observed between overall fit and prediction bias. Models with higher R^2 and lower RMSE, such as CNN2D-Uniclass, tend to overestimate CF, resulting in higher MBE values. Conversely, models with lower bias, particularly those using the Physics-based strategy, show reduced R^2 and increased RMSE. This indicates that while some pixel selection strategies improve general alignment with the reference, they may introduce systematic offsets, highlighting the need to balance accuracy and bias depending on the intended application.

Correspondingly, Table 4.9 shows the regression metrics for each CNN model, this time excluding pixels with cloud-free and cloud-full conditions compared again with the PACO baseline.

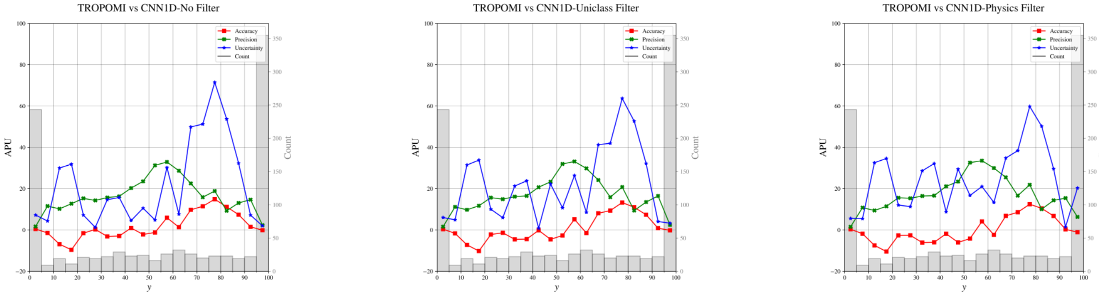
Classifier		R^2	RMSE	MBE
PACO		0.49	4.39	0.55
CNN1D	*No-Filter*	0.46	4.45	2.35
	Uniclass	0.44	4.49	1.20
	Physics	0.42	4.53	0.30
CNN2D	*No-Filter*	0.49	4.40	2.25
	Uniclass	0.49	4.40	2.48
	Physics	0.46	4.45	1.09
CNN3D	*No-Filter*	0.44	4.49	0.68
	Uniclass	0.45	4.47	1.26
	Physics	0.46	4.46	1.57

Table 4.9: Regression metrics. Testing dataset include non full-cloud or clear-sky pixels ($1 \leq \text{CF} \leq 99$), under 180 s between captures.

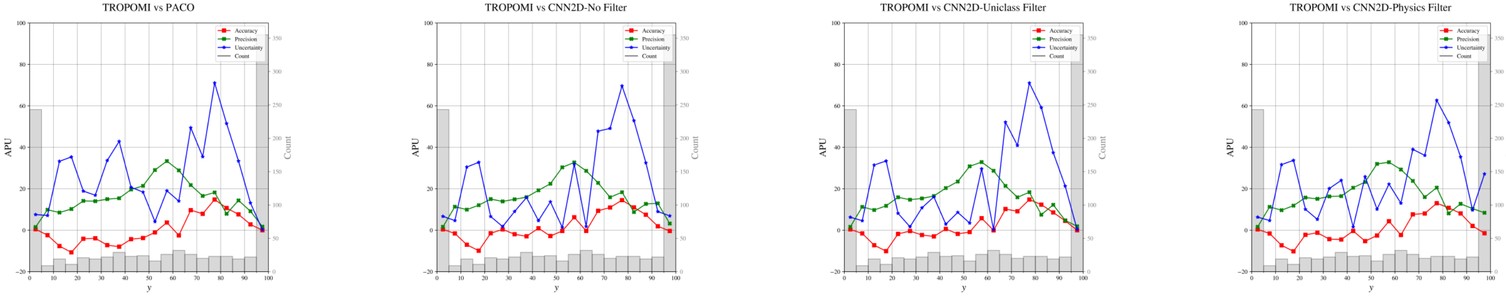
.

When excluding the cloud-free and cloud-full pixels, the performance of all classifiers decreases, especially in terms of R^2. For example, the baseline PACO drops from 0.90 to 0.49, a decrease of -0.41, highlighting the strong influence of the discarded pixels on the overall metrics. In this range of partial cloud cover, differences between models become more noticeable. CNN models tend to overestimate cloud fraction, as seen in the higher MBE values compared to PACO. Among the CNNs, CNN2D still shows the closest performance to PACO in terms of R^2, but with a larger bias.

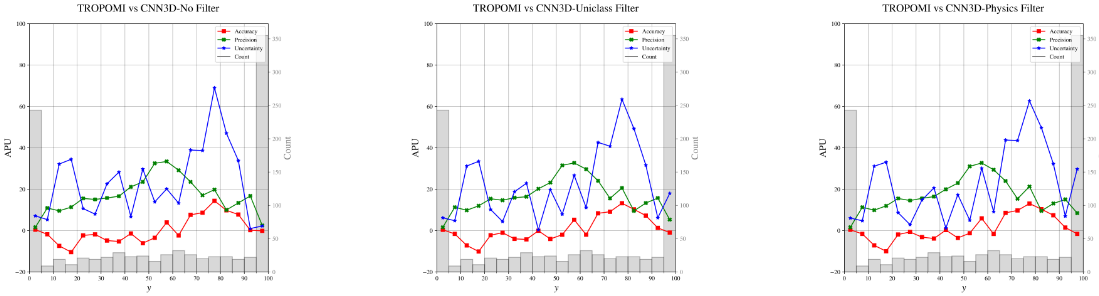
Figures 4.12 and 4.13 show the APU plots and 2D regression histograms for all pixels, comparing each tested CNN with the PACO baseline. These plots allow the evaluation of predictions across the full CF range, as they separate the densely populated bins corresponding to cloud-free and cloud-full conditions.



(a) CNN1D No Filter (b) CNN1D Uniclass (c) CNN1D Physics

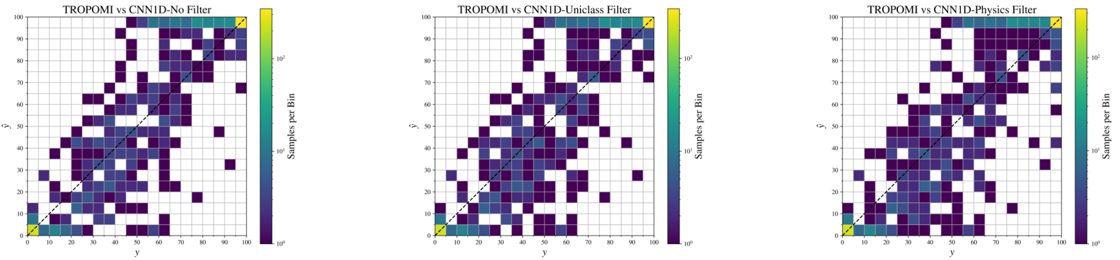


(d) PACO (e) CNN2D No Filter (f) CNN2D Uniclass (g) CNN2D Physics

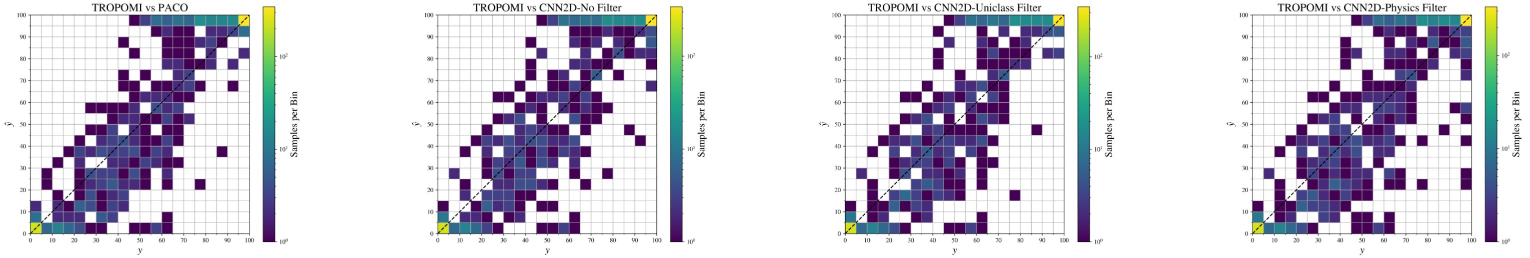


(h) CNN3D No Filter (i) CNN3D Uniclass (j) CNN3D Physics

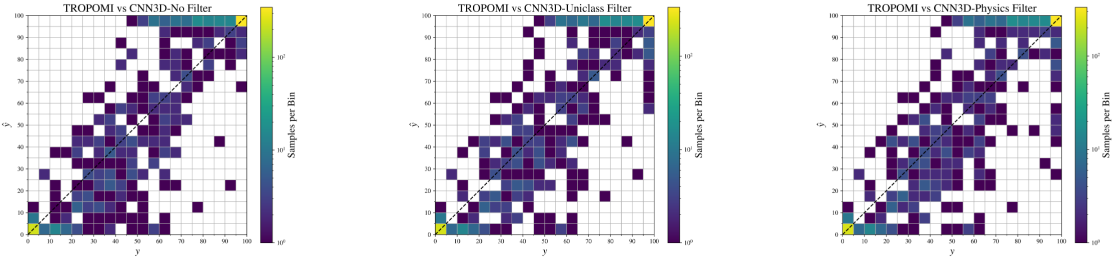
Figure 4.12: APU plots comparing TROPOMI with model predictions for all architectures and filter types.



(a) CNN1D No Filter (b) CNN1D Uniclass (c) CNN1D Physics



(d) PACO (e) CNN2D No Filter (f) CNN2D Uniclass (g) CNN2D Physics



(h) CNN3D No Filter (i) CNN3D Uniclass (j) CNN3D Physics

Figure 4.13: 2D histograms comparing TROPOMI with model predictions for all architectures and filter types.

From the APU plots shown in Figure 4.12, the three metrics (A, P, and U) are evaluated across all CF bins for each tested CNN model and compared with the PACO baseline.

- **Low CF (0–20%)**
  Across all classifiers, including PACO, the APU metrics are fairly consistent in this range. A is close to zero in cloud-free sky but shows underestimation between $5 \leq \text{CF} \leq 20$, while both P and U remain relatively low and stable. There are no major deviations between models in this region, indicating that low CF conditions are generally predicted with similar accuracy across architectures and filtering strategies.

- **Mid CF (20–60%)**
  In this range, the differences between classifiers become more apparent. CNN2D models, particularly with the *No filter* and *uniclass* strategies (Figures 4.12e and f), show a clear improvement in A compared to PACO and other CNNs. The bias remains consistently closer to zero, while U and P are comparable or slightly reduced. In contrast, CNN1D and CNN3D models show less stability, with A values diverging further from zero and greater variability in P.

- **High CF (60–100%)**
  This range shows the most evident performance degradation for some models. Specifically, CNN3D with the *uniclass* filter (Figure 4.12i) and all CNNs using the *physics filter* (Figures 4.12c, g, j) exhibit significantly worse performance in terms of P and U in the extreme bin of cloud-full sky. In the rest of the mentioned range, A sharply deviates from zero (indicating a large bias), and both P and U reach their highest values. These trends suggest that these configurations are less reliable in densely cloudy scenes. Additionally, around CF = 75%, U reaches its peak, while A remains systematically high, indicating overestimation of clouds in that range.

The APU analysis shows that while performance is generally stable at low CF, clear improvements in A are achieved by CNN2D models in the 20–60% CF range. However, high CF bins reveal major weaknesses in CNN3D with the *uniclass* filter and all CNNs using the *physics filter*, where both error and variability increase sharply. These results support the robustness of CNN2D, particularly with the *uniclass* filter.

Although the differences are not as noticeable as in the APU plots, the mentioned deviation in the high CF range is visible in the 2D histogram comparisons shown in Figures 4.13-c, g, and j, where the *physics filter* underestimates clouds. Additional variations along the diagonal are present, but no specific patterns are clearly observed.

The rest of regression plots and differences can be consulted in Appendix C.

4.2.5 Qualitative Analysis

In this section, the classification maps are evaluated qualitatively to allow subjective comparison with the reference from TROPOMI, highlighting the improvements reflected or not on by comparing with the reference data, or conversely, misclassifications or systematic deviations in either the classification or the reference CF.

For all the following presented scene results and comparisons between the baseline from PACO and the trained CNNs, the remaining models perform comparably and are thus omitted to avoid redundancy. Any exceptions will be explicitly indicated. In addition, the presented scenes were chosen without any particular preference including all pixel selection strategies, or, if that was the case, with the aim of selecting those with more pronounced differences compared to PACO.

4.2.5.1 Low-Mid Cloud Fraction

In this subsection, some improvements that are lost when averaging cloud coverage over the TROPOMI pixel footprint are highlighted.

For instance, Figure 4.14 presents a case of coastal white sand misclassified as clouds by PACO, which is mostly correctly classified by CNN1D-*physics*. The coastal area is small enough that it does not significantly affect the CF, although the correction constitutes a clear improvement when evaluated qualitatively.

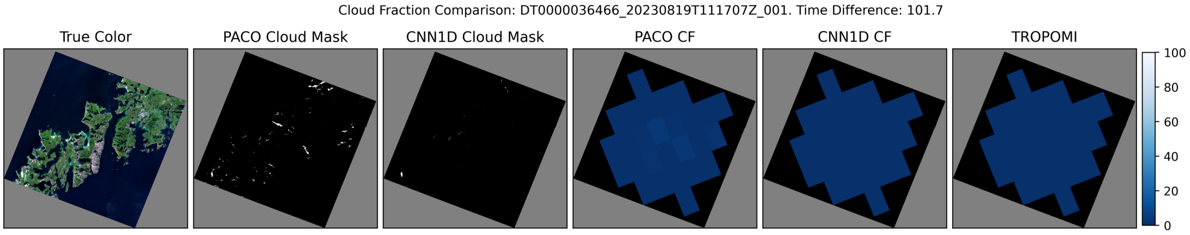


Figure 4.14: Cloud masks and CF comparison between PACO and CNN1D-*physics*. Image: EnMAP DT-36466, tile-001, captured at 2023/08/19.

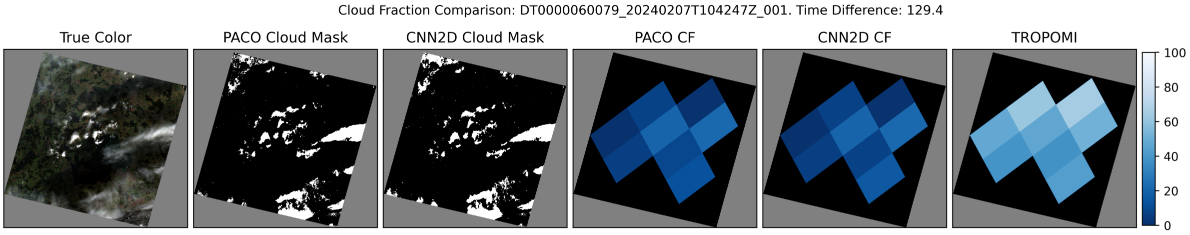
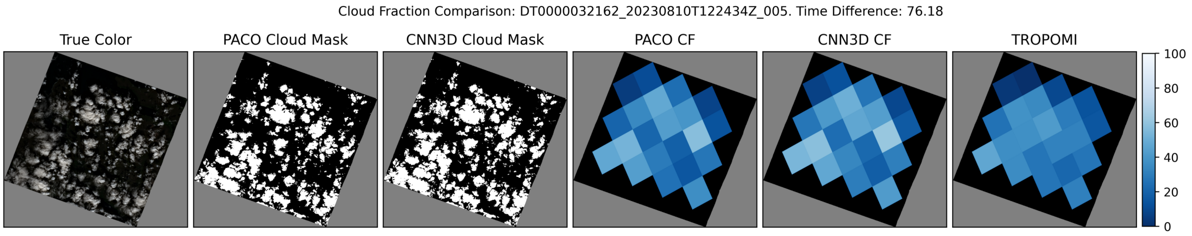


Figure 4.15: Cloud masks and CF comparison between PACO and CNN2D-*uniclass*. Image: EnMAP DT-60079, tile-001, captured at 2024/02/07.

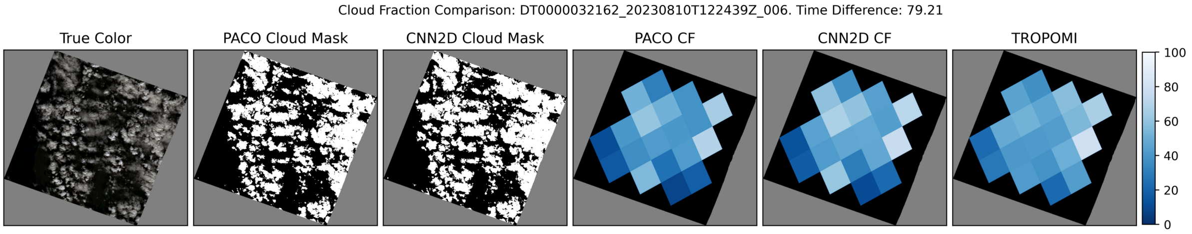
An example of reference deviation is shown in Figure 4.15, where isolated clouds are visible in the EnMAP image but correspond to high CF values in the reference pixels of TROPOMI. It is observed that most of the clouds are correctly labeled by both classifiers, showing similar cloud masks, although CF overestimation is observed in the reference. When evaluated qualitatively, a substantial increase in masked clouds by CNN2D to match the reference CF would be interpreted as an overestimation.

Such difference between CF comparisons is not well understood. Some differences are expected given the inherent uncertainties of OCRA algorithm used by TROPOMI, together with the time difference between captures, but specifically in the case shown in Figure 4.15, the gap is larger than expected. One hypothesis is that the DB used by OCRA relies on background reflectances derived from very dark, dense vegetation with low reflectance. The presence of isolated clouds together with vegetation stress, indicated by the yellowish tone in the True Color quick-look, raises the background reflectance and may contribute to the CF overestimation by OCRA.

A counter example is shown in Figure 4.16a, where an EnMAP image with a low-reflectance background shows less differences between the classifiers and the reference.



(a) CNN3D-*physics*, tile-005.



(b) CNN2D-*physics*, tile-006.

Figure 4.16: Cloud masks and CF comparison between PACO and CNN-based models. EnMAP DT-32162, captured on 2023/08/10.

The difference between cloud masks from PACO and CNN3D is small due to the high contrast between the background and the highly reflective clouds. Contrary to the example shown in Figure 4.15, there is a slight bias toward overestimation relative to the reference. An interesting aspect to look is the neighboring tile of 4.16a, presented in Figure 4.16b, where the cloud masks from PACO and CNN2D-*physics* are compared this time. It can be seen that the classification performance between CNN3D and CNN2D appear nearly identical. This trend is present in most of the images in the testing dataset, where the changes between classifiers are marginal.

As shown in Figure 4.16, EnMAP captures data-takes composed of multiple tiles. Consequently, many pixels within a single scene share similar characteristics and conditions, as they originate from the same geographic location and moment. Although the testing dataset includes 105 EnMAP scenes, these correspond to only 18 distinct TROPOMI overpasses. This spatial and temporal alignment results in clusters in the regression plots, as groups of pixels exhibit similar conditions and thus similar predictive behavior.

The presented examples suggest that changes in background that affect the reflectances can influence the CF retrieved by OCRA. This is also noted in OCRA's ATBD [146] and during the validation of its cloud products [128], where a continuous update of the background DB is described, along with expected higher uncertainties over highly reflective surfaces (e.g., over areas covered in fresh snow). Unfortunately, from the 105 EnMAP scenes matching overpasses of Sentinel-5P, none of them corresponds to snow surfaces, or, if any exist, they are completely covered by clouds.

4.2.5.2 High Cloud Fraction

In this subsection, a similar analysis to that in Subsection 4.2.5.1 is conducted; however, the comparisons in the high CF range exhibit larger discrepancies, with the previously discussed issues related to the comparison between GCF and RCF (see Subsection 4.2.2) becoming more pronounced.

An average case is presented in Figure 4.17, showing good agreement between the predicted and reference CF, both in spatial distribution and CF values. This example is included to illustrate a typical scenario with a low time difference between captures.

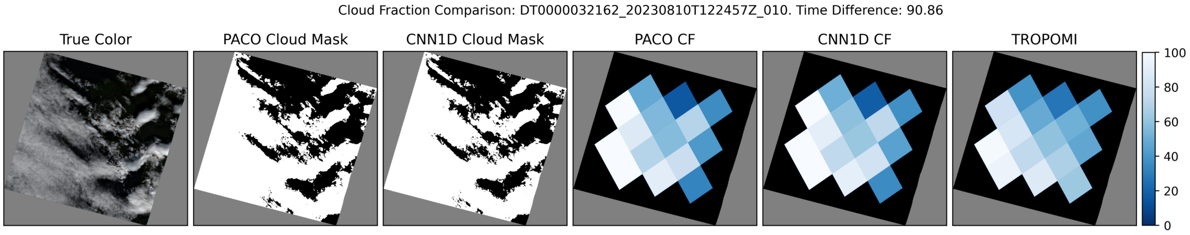


Figure 4.17: Cloud masks and CF comparison between PACO and CNN1D-*uniclass*. Image: EnMAP DT-38162, tile-010, captured at 2023/08/10.

A problem discussed in Subsection 4.2.2 is that GCF tends to produce higher values when clouds exhibit low reflectance. This effect is illustrated in Figure 4.18, where semi-transparent clouds on the right side of the scene show low reflectance, resulting in a lower RCF in the reference data. Although CNN3D-*uniclass* underdetects clouds in the corner of the scene, these clouds fall outside the area covered by any corresponding TROPOMI pixel, and therefore do not affect the computed CF comparison.

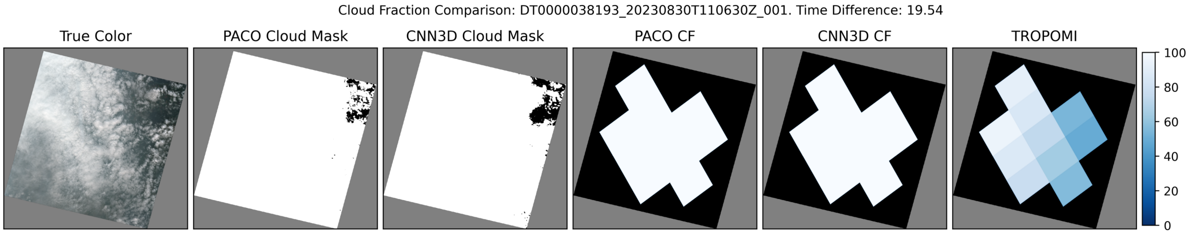


Figure 4.18: Cloud masks and CF comparison between PACO and CNN3D-*uniclass*. Image: EnMAP DT-38193, tile-001, captured at 2023/08/30.

Conversely, Figure 4.19 shows reduced detection of semi-transparent clouds in the bottom-left region of the scene. In this case, the reference CF is higher than both the CF retrieved by PACO and that predicted by CNN3D-*uniclass*, highlighting an overestimation by the reference under these conditions.

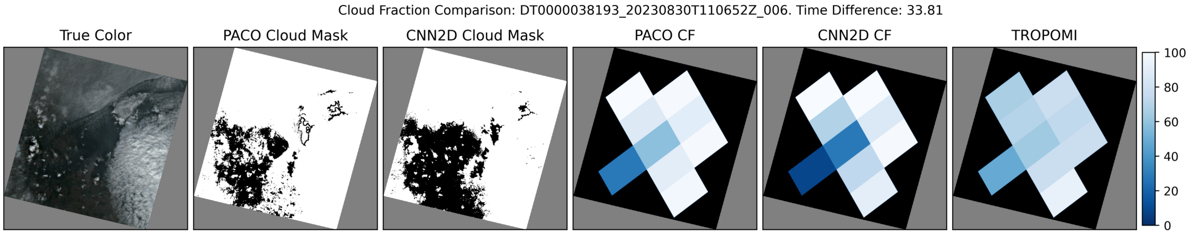


Figure 4.19: Cloud masks and CF comparison between PACO and CNN2D-*physics*. Image: EnMAP DT-38193, tile-006, captured at 2023/08/30.

The most noticeable error in the proposed approach is illustrated in Figure 4.20 and discussed in Subsection 4.2.4. This case highlights a significant underestimation of clouds by the *physics filter* in scenes with high CF. A substantial portion of the scene (approximately one-third) appears fully cloud-covered but is missed by CNN2D, leading to a lower performance in this range compared to PACO.

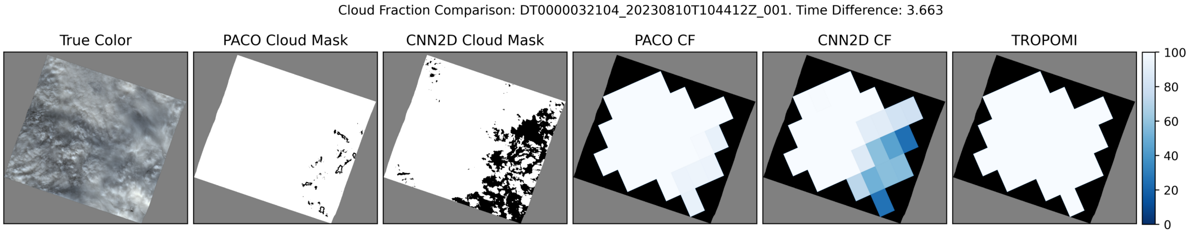
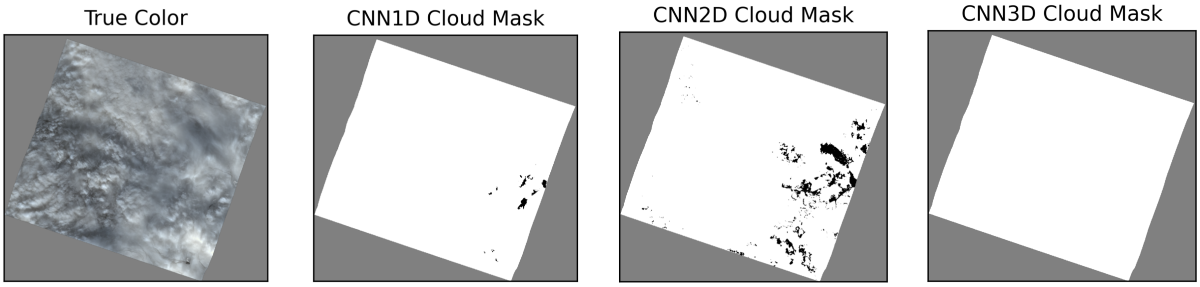


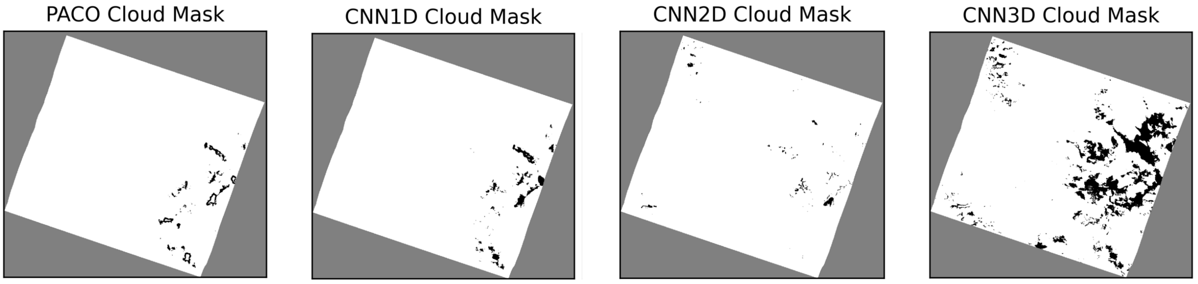
Figure 4.20: Cloud masks and CF comparison between PACO and CNN2D-*physics*. Image: EnMAP DT-32104, tile-001, captured at 2023/08/10.

To illustrate model performance on this challenging scene, Figure 4.21 presents the cloud masks predicted by all trained models. The true-color image (a) reveals widespread cloud coverage, which is mostly correctly classified by the PACO reference mask (e). The *physics filter* (PF) models (i–k) perform the worst, consistently miscclassifiying the cloud-covered area. While CNN3D-*uniclass* (h) and CNN2D-*no filter* (c) also miss a portion, their predictions better approximate the reference.

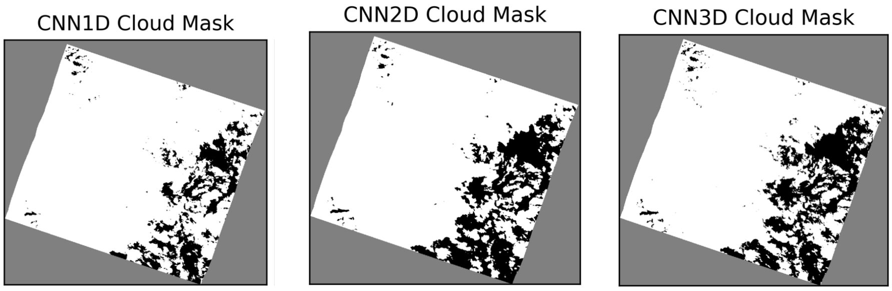
Apart from the presented scene, most of the evaluated cases show good performance across the CNN models, with cloud predictions generally aligning well with both the true-color imagery and showing no major differences from the PACO cloud mask.



(a) True color (b) CNN1D NoF (c) CNN2D NoF (d) CNN3D NoF



(e) PACO (f) CNN1D UF (g) CNN2D UF (h) CNN3D UF



(i) CNN1D PF (j) CNN2D PF (k) CNN3D PF

Figure 4.21: Qualitative comparison of cloud predictions by CNN models using different pixel selection strategies. Rows represent filter strategies (*No Filter, Uniclass, Physics*), and columns correspond to CNN architectures (1D, 2D, 3D). The first column shows the true-color image and the PACO cloud mask for reference.

4.3 Discussion

The results presented in the previous sections demonstrated the potential and limitations of the proposed classification framework when applied to both multispectral and hyperspectral imagery. This section interprets those results in detail, offering a critical analysis of the factors influencing model performance and the broader implications of the findings. Particular attention is given to how pixel selection strategies affect classification, the challenges of generalizing across diverse conditions, and the trade-offs between spectral and spatial feature extraction. Additionally, this discussion contextualizes the regression-based validation for hyperspectral imagery and explores the impact of spatial resolution mismatches and temporal uncertainties. Finally, directions for future improvements, such as explainable AI, physics integration, and sensor scalability, are discussed to guide further research and operational deployment.

4.3.1 Discussion on Results for Multispectral Imagery

This work proved that by selecting high-quality pixels from a given multi-spectral DB with reference labels obtained by thresholding spectral indices from the atmospheric corrector PACO, classification performance can be improved on independent test scenes by +3.3pp on TD1 and +18.3pp on TD2 in terms of $nMCC$ (Table 4.1) compared with the operational baseline *hcw* classification map from PACO. Notably, the greatest improvement was observed for the pixel selection using the *uniclass filter* for CNN2D and using the *physics filter* for CNN1D, which both discard high uncertainty pixels for composing the training set. This improvement is particularly evident in cases where pixels are mixed at object borders or where the spectrum is similar, such as the *shadow* and *clear* classes, with improvements of 4-20.2pp and 1.1-22.7pp, respectively, thanks to the advanced feature extraction capabilities of the deep learning models tested in this work. In addition, the rest of classes performed acceptably, with moderate improvements of $\leq$ 6.1pp, but losing in the snow class for TD1 up to 7.4pp.

The presented experiments with different training dataset selection strategies demonstrated significant changes in the behavior of the implemented classifier, and even if it is not reflected in the $nMCC$ metric, the qualitative analysis demonstrated a clear improvement. For example, CNN2D results for TD1 appear similar in Table 4.1, but a noticeable improvement can be observed in Fig. 4.2 for CNN2D-*uniclass filter*. The improvements are more remarkable for TD2, which contains challenging classification cases, where the *physics filter* supports the spectral feature extraction of CNN1D, while the *uniclass filter* benefits the joint spatial-spectral feature extraction of CNN3D.

Despite the significant feature extraction capabilities of CNNs, their performance heavily depends on the quality of the training set and its reference labels. The variety of testing datasets evaluated in this work highlighted that the training dataset is limited by the lack of representation for all possible subclasses under diverse atmospheric conditions, illumination angles, and noisy labels. Consequently, fully representing all classes for previously unseen pixels remains a significant challenge.

Although qualitative analysis is subjective, some regions demonstrated clearly visible improvements that were not included in the testing datasets, potentially concealing the true classification performance of the proposed models.

4.3.2 Discussion on Results for Hyperspectral Imagery

The obtained results of trained models on hyperspectral imagery from the EnMAP sensor with the proposed testing methodology comparing both GCF and RCF have not shown significant improvements in cloud classification over PACO's cloud masking.

However, some small improvements were identified during qualitative assessment of the cloud mask, but these are not reflected in regression metrics due to differences in spatial resolutions.

There are several hypotheses for this behavior when compared to the results obtained for the multispectral case discussed earlier, where most improvements were identified in small regions and in challenging classification cases. The improvements were smaller for TD1, which is composed of average scenarios, whereas larger improvements were observed for TD2. From the 105 scenes that compose the testing dataset, none appear to be challenging classification cases for cloud masking.

In addition, the greatest improvements for the multispectral case were identified for the *shadows* and *clear* classes. Given the limitation of the proposed testing methodology to validate only the cloud mask, the classifier was only trained for binary classification. As a result, the benefits of multi-class classification in a single DL model for better defining decision boundaries are reduced.

On the other hand, the lack of available pixels across the full CF range limits the global metrics (R^2, RMSE, MBE), and this limitation depends on the number of matching overpasses between the two missions, which can be improved as operational time increases.

The systematic differences between RCF and GCF across different CF ranges highlight the importance of using APU plots to assess performance for each bin. In addition, regarding completeness of the testing dataset, quantifying how uncertainty grows as the time difference increases is an interesting topic for future work, as many sources of information could help estimate the comparability between both cloud products.

4.3.3 Dataset Coverage and Computational Complexity

To further improve this approach, additional sources of information should be incorporated to reduce label noise and enhance the representation of each class. From a global application perspective, this remains challenging due to the computational and spatial complexity of training deep learning models. This highlights the need for advanced data selection strategies and novel training approaches to fully exploit the potential of multispectral imagery, and by extension, hyperspectral imagery. In addition, incorporating supplementary features can improve data characterization, as deep learning models are capable of learning patterns from diverse sources. For instance, viewing and solar angles could help distinguish between cloud or topographic shadows and clear-sky land under low illumination conditions. This differentiation can be further supported by spatial information. For example, if a shadowed region is adjacent to a detected cloud and the solar-viewing geometry is consistent, it is likely that the shadow belongs to that cloud.

We observed that CNN1D, with only 77 k parameters, achieves outstanding performance compared with CNN2D (155 k parameters) and CNN3D (305 k parameters), highlighting the importance of pure-spectral feature extraction. In our prototype, the wall-clock time (per epoch) of CNN2D and CNN3D is nearly identical due to the custom data generator, which dominates total runtime by creating batches of spatial patches, hiding the theoretical differences on computational complexity between 2D and 3D convolutional layers [147]. We leave an in-depth computational cost analysis for future work because of the current implementation's limitations.

Since we aimed to avoid introducing any information from the testing dataset into the training process and to study methods for obtaining a generalized model applicable to any case, we observed that increasing the number of training pixels or the number of training epochs, despite the validation set improving slightly with each epoch, does not result in better performance for the testing datasets. This behavior is likely because enlarging the training set also increases the likelihood of outliers or non-representative samples. Additionally, by introducing highly correlated pixels corresponding to average cases, the training process may overfit the original model, replicating its behavior, and losing the generalization that comes from a sparse input. However, these are only hypotheses and should be further studied.

To strengthen domain generalization, the sample size must grow, but instead of including more pixels within the same limited number of scenes, this should be done by selecting an extensive set of additional scenes spanning diverse locations, biomes, seasons, and atmospheric conditions. Capturing a wider spectroscopic variability should yield a more representative training set and, therefore, improved generalization.

4.3.4 Testing Protocols

The per-class classification analysis of the multispectral case reveals the biases of the classifier. Since each dataset provides slightly different conclusions, it is evident that the independently and manually labeled testing datasets have varying class definitions or are not uniformly distributed across all subclasses. Achieving such uniformity is nearly impossible with manual labeling due to the significant manpower required.

The approach presented in this thesis for the testing phase of the hyperspectral case is the first attempt at performing classification validation for the hyperspectral sensor

EnMAP using external sources of information, with the aim of validating it in terms of CF, a physical parameter. Given that GCF from EnMAP and RCF from TROPOMI are not directly comparable, a harmonized calibration between them is currently lacking. In addition, the difference between spatial resolutions makes it difficult to compare small changes in classification.

In addition, more robust testing methods for automatically retrieving highly sensitive reference data should be developed to achieve completeness and avoid biases, as well as by incorporating other missions specifically designed for the atmospheric phenomenon study that retrieve physical parameters for comparison.

Furthermore, a harmonized definition based on physical parameters should be established for each class, considering a compilation of the requirements from a wide range of applications that make use of the distributed L2 products. This would enable comparisons across classifiers and allow the interpretation of the generalization for global-scale applications.

4.3.5 Future Directions: XAI and PINNs

Future work will consider the use of Explainable Artificial Intelligence (XAI) [148, 149, 150, 151, 152], to identify the most important features for classification in each class, since the spectral characteristics are basic features for pixel classification. Additionally, the potential incorporation of well-studied physics models from other remote sensing areas will be explored through the use of Physics-Informed Neural Networks (PINNs) [153, 154]. This integration aims to build robust models, not relying only on a statistical perspective, addressing the issue of sparse subclass representation while still leveraging the complex spatial and spectral feature extraction capabilities of deep learning models.

4.3.6 Scalability to Other Sensors

Regarding the scalability of this work, while this study focuses on Sentinel-2 and EnMAP imagery, the proposed methodology could be applied to other multispectral or hyperspectral sensors (by training a model for each sensor), as long as a pre-classification input is available for training. While the current implementation is built using PACO, which supports Sentinel-2 [9], Landsat-8 [6, 7], EnMAP [10, 11], and DESIS [5, 34], the physical rules used for pixel selection are sensor-independent and derived from fundamental reflectance-based principles. In cases where a different masking algorithm is used to generate training masks, the same logic can be adapted to new class definitions. Further evaluation on additional optical remote sensing missions with global-scale implementation is left for future work.

Chapter 5

Conclusions

This thesis investigated the use of CNN1D, CNN2D, and CNN3D architectures to improve classification products in the context of AC for multispectral and hyperspectral imagery. The approach adopted in this thesis differs from common classification strategies, where training and testing sets typically originate from the same dataset and, consequently, share the same labeling source, often based in manual annotation. In this thesis, the models were trained using PACO's masking products, combined with spatial information and additional features, addressing the general objective of generating improved classification results using DL in operational EO systems. This strategy reduces the need for manual labeling prior to training, making it particularly well-suited for implementation during the commissioning phase of upcoming remote sensing missions, when ground-truth labels are scarce or unavailable.

The particular objective of extracting spatial features from the spectral image was fulfilled by testing CNN2D and CNN3D architectures. Results show consistent improvements in classification accuracy over the baseline PACO "hcw" maps, especially in scenes containing mixed pixels or spectral ambiguity. In the multispectral case, CNN2D and CNN3D models outperformed CNN1D in spatially complex areas, confirming the benefit of spatial context. This included the extra inputs; topographic elevation from a DEM and pixel-wise illumination geometry, providing context beyond the spectral signature.

Evaluation was performed using independent testing datasets. For multispectral data, accuracy was assessed through nMCC on more and less conservative labeled populations, both independently and manually labeled. Hyperspectral results were validated using regression metrics and APU plots based on CF comparisons with TROPOMI products. This fulfilled the objective of testing model performance on unseen scenes using independent data sources.

In terms of analyzing strengths and limitations, the CNNs demonstrated strong performance in identifying classes such as *shadow* and *clear*, particularly when trained with filtered labels that reduce classification uncertainty. However, the models were sensitive to the quality and representativeness of the training data. While CNN1D was effective in cases with clean spectral features, spatial models better handled ambiguity at class boundaries. These observations supported a broader understanding of how different architectures behave under varying scene conditions.

A key contribution of this thesis was the identification of weaknesses in standard validation protocols. Conventional metrics such as R^2 and RMSE were found to be biased by overrepresented cloud-free or fully clouded pixels. In response, this work proposed evaluation methods based on physical parameters and adopted robust metrics for both multi-class and binary classification cases, such as nMCC, as a more reliable alternative to the widely used OA. The validation strategies included comparing geometric and radiometric CF, and introduced the use of APU plots to evaluate model behavior across the full CF range. This addressed the objective of proposing more robust, bias-aware testing strategies for pixel-based classification. Furthermore, a harmonized definition of classes is necessary for testing, allowing testing datasets to provide a more generalized interpretation of results—especially when based on physical parameters characteristic of the mask itself (e.g., snow grain size, cloud altitude, albedo, among others). The techniques developed in this thesis contribute to defining guidelines for the future development of robust and scalable methodologies for DL approaches in optical remote sensing missions with global-scale application.

Addressing the case of recently launched missions, although this methodology requires data for training, the imagery obtained during the commissioning phase is often sufficient to set up an operational model. The reduced time frame makes it very challenging to manually label enough data to obtain a generalized model; therefore, there is a clear need to retrieve high-quality labels automatically, such as through the strategies presented in this work.

Finally, this study contributes a framework for training DL-based classifiers without manual labeling, using filtering rules to automatically compile high-quality datasets. These strategies are applicable to operational remote sensing missions and are scalable to new sensors and processing chains. The findings emphasize the importance of physically-informed label generation, spatial-spectral learning, and validation protocols tailored to the specific challenges of AC and cloud masking.

The following points outline the proposed future work based on the findings and limitations identified in this thesis. They aim to enhance the scalability, robustness, and physical interpretability of DL-based classification approaches in the context of remote sensing and AC.

- Automatic compilation of large-scale training datasets with high-quality labels that capture both “safe” (conservatively annotated) pixels and difficult, outlier cases (e.g., those present in TD2).
- The development of new testing and validation methods based on physical parameters, and the harmonization of class definitions accordingly.

- The integration of advanced training techniques, such as the use of Explainable Artificial Intelligence (XAI) and Physics-Informed Neural Networks (PINNs).
- The implementation of the proposed methodology for hyperspectral missions using more classes, where the high number of spectral bands reduces the effectiveness of traditional threshold-based methods, and spatial-spectral learning can offer a more scalable alternative.

This page is intentionally left blank.

Chapter 6

Bibliography

[1] W. Emery and A. Camps, *Introduction to satellite remote sensing: atmosphere, ocean, land and cryosphere applications.* Elsevier, 2017.

[2] M. Urai, S. Tsuchida, S. Yamamoto, T. Tachikawa, A. Iwasaki, M., and J. Ishii, "Initial Onboard Calibration Results of the {HISUI} Hyperspectral Sensor," in *Proceedings of 2021 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021, pp. 1608–1610.

[3] D. S. Connelly, D. R. Thompson, N. M. Mahowald, L. Li, N. Carmon, G. S. Okin, and R. O. Green, "The {EMIT} mission information yield for mineral dust radiative forcing," *Remote Sensing of Environment*, vol. 258, 2021.

[4] S. Cogliati, F. Sarti, L. Chiarantini, M. Cosi, R. Lorusso, E. Lopinto, F. Miglietta, L. Genesio, L. Guanter, A. Damm, S. Pérez-López, D. Scheffler, G. Tagliabue, C. Panigada, U. Rascher, T. P. F. Dowling, C. Giardino, and R. Colombo, "The {PRISMA} imaging spectroscopy mission: overview and first performance analysis," *Remote Sensing of Environment*, vol. 262, 2021.

[5] K. Alonso, M. Bachmann, K. Burch, E. Carmona, D. Cerra, R. de los Reyes, D. Dietrich, U. Heiden, A. Hölderlin, J. Ickes, U. Knodt, D. Krutz, H. Lester, R. Müller, M. Pagnutti, P. Reinartz, R. Richter, R. Ryan, I. Sebastian, and M. Tegler, "Data products, quality and validation of the dlr earth sensing imaging spectrometer (desis)," *Sensors (Switzerland)*, vol. 19, p. 4471, 10 2019. [Online]. Available: www.mdpi.com/journal/sensors

[6] J. A. Barsi, K. Lee, G. Kvaran, B. L. Markham, and J. A. Pedelty, "The Spectral Response of the Landsat-8 Operational Land Imager," *Remote Sensing 2014, Vol. 6, Pages 10232-10251*, vol. 6, no. 10, pp. 10 232–10 251, 10 2014. [Online]. Available: https://www.mdpi.com/2072-4292/6/10/10232/htmhttps://www.mdpi.com/2072-4292/6/10/10232

[7] E. Vermote, C. Justice, M. Claverie, and B. Franch, “Preliminary analysis of the performance of the landsat 8/oli land surface reflectance product,” *Remote Sensing of Environment*, vol. 185, pp. 46–56, 2016, landsat 8 Science Results. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425716301572

[8] J. G. Masek, M. A. Wulder, B. Markham, J. McCorkel, C. J. Crawford, J. Storey, and D. T. Jenstrom, “Landsat 9: Empowering open science and applications through continuity,” *Remote Sensing of Environment*, vol. 248, p. 111968, 10 2020.

[9] M. Drusch, U. Del Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort, A. Meygret, F. Spoto, O. Sy, F. Marchese, and P. Bargellini, “Sentinel-2: ESA’s Optical High-Resolution Mission for GMES Operational Services,” *Remote Sensing of Environment*, vol. 120, pp. 25–36, 5 2012.

[10] L. Guanter, H. Kaufmann, K. Segl, S. Foerster, C. Rogass, S. Chabrillat, T. Kuester, A. Hollstein, G. Rossner, C. Chlebek, C. Straif, S. Fischer, S. Schrader, T. Storch, U. Heiden, A. Mueller, M. Bachmann, H. Mühle, R. Müller, M. Habermeyer, A. Ohndorf, J. Hill, H. Buddenbaum, P. Hostert, S. der Linden, P. J. Leitão, A. Rabe, R. Doerffer, H. Krasemann, H. Xi, W. Mauser, T. Hank, M. Locherer, M. Rast, K. Staenz, and B. Sang, “The EnMAP Spaceborne Imaging Spectroscopy Mission for Earth Observation,” *Remote Sensing*, vol. 7, no. 7, pp. 8830–8857, 2015. [Online]. Available: https://www.mdpi.com/2072-4292/7/7/8830

[11] T. Storch, H.-P. Honold, S. Chabrillat, M. Habermeyer, P. Tucker, M. Brell, A. Ohndorf, K. Wirth, M. Betz, M. Kuchler *et al.*, “The enmap imaging spectroscopy mission towards operations,” *Remote Sensing of Environment*, vol. 294, p. 113632, 2023.

[12] R. de Los Reyes, M. Langheinrich, P. Schwind, R. Richter, B. Pflug, M. Bachmann, R. Müller, E. Carmona, V. Zekoll, and P. Reinartz, “Paco: Python-based atmospheric correction,” *Sensors (Basel, Switzerland)*, vol. 20, p. 1428, 3 2020. [Online]. Available: www.mdpi.com/journal/sensors

[13] P. Shippert, “Why use hyperspectral imagery?” 2004.

[14] J. Nieke, L. Despoisse, A. Gabriele, H. Weber, H. Strese, N. Ghasemi, F. Gascon, K. Alonso, V. Boccia, B. Tsonevska, P. Choukroun, G. Ottavianelli, and M. Celesti, “The copernicus hyperspectral imaging mission for the environment (chime): an overview of its mission, system and planning status,” in *Sensors, Systems, and Next-Generation Satellites XXVII*, S. R. Babu, A. Hélière, and T. Kimura, Eds., vol. 12729. SPIE, 2023, p. 1272909. [Online]. Available: https://doi.org/10.1117/12.2679977

[15] M. Celesti, M. Rast, J. Adams, V. Boccia, F. Gascon, C. Isola, and J. Nieke, "The copernicus hyperspectral imaging mission for the environment (chime): Status and planning," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2022-July, pp. 5011–5014, 2022.

[16] D. R. Thompson, R. Basilio, I. Brosnan, K. Cawse-Nicholson, K. D. Chadwick, L. Guild, M. Gierach, R. O. Green, S. Hook, S. D. Horner, G. Hulley, R. Kokaly, C. E. Miller, K. R. Miner, C. Lee, D. Limonadi, J. Luvall, R. Pavlick, B. Phillips, B. Poulter, A. Raiho, K. Reath, S. S. Uz, A. Sen, S. Serbin, D. Schimel, P. Townsend, W. Turner, K. Turpie, and T. S. Team, "Ongoing progress toward nasa's surface biology and geology mission," in *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 5007–5010.

[17] R. A. Schowengerdt, *Remote Sensing*, 3rd ed. Tucson, Arizona: Elsevier Inc., 2007.

[18] M. Borengasser, W. S. Hungate, and R. L. Watkins, *Hyperspectral remote sensing : principles and applications.* Indiana, U.S.A.: CRC Press, 2008.

[19] R. Curley, "Electromagnetic spectrum — definition, diagram, and uses — britannica," 2 2009. [Online]. Available: https://www.britannica.com/science/electromagnetic-spectrum

[20] G. Elert, "Electromagnetic spectrum." [Online]. Available: https://physics.info/em-spectrum/

[21] R. Paschotta, "Infrared light," RP Photonics Encyclopedia. [Online]. Available: https://www.rp-photonics.com/infrared_light.html

[22] T. Bruno and P. Svoronos, *CRC Handbook of Fundamental Spectroscopic Correlation Charts.* CRC Press, 2005. [Online]. Available: https://books.google.de/books?id=FgjHjhCh5wsC

[23] "Enmap hsi instrument specification." [Online]. Available: https://www.enmap.org/data/doc/EnMAP_Specs.pdf

[24] R. de los Reyes, M. Langheinrich, and M. Bachmann, "Level 2a processor (atmospheric correction over land) atbd," Dec 2023. [Online]. Available: https://www.enmap.org/data/doc/EN-PCV-TN-6007_Level_2A_Processor_Atmospheric_Correction_Land.pdf

[25] A. Berk, *MODTRAN 5.2.0 user's manual*, 2008. [Online]. Available: ftp://ftp.pmodwrc.ch/stealth/132250_claus/MODTRAN5/Manual/MODTRAN(R)5.2.0.0.pdf

[26] L. Guanter, H. Kaufmann, S. Förster, A. Brosinsky, H. Wulf, M. Bochow, N. Boesche, M. Brell, H. Buddenbaum, S. Chabrillat *et al.*, “Enmap science plan,” 2016.

[27] R. C. Gonzalez and R. E. Woods, *Digital image processing*, 3rd ed. Pearson, 2008.

[28] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” pp. 455–500, 2009.

[29] R. Richter and D. Schläpfer, “Atmospheric and topographic correction (atcor theoretical background document),” pp. 503–564, 2019.

[30] T. R. Loveland and J. L. Dwyer, “Landsat: Building a strong future,” *Remote Sensing of Environment*, vol. 122, pp. 22–29, 2012.

[31] M. A. Wulder, D. P. Roy, V. C. Radeloff, T. R. Loveland, M. C. Anderson, D. M. Johnson, S. Healey, Z. Zhu, T. A. Scambos, N. Pahlevan *et al.*, “Fifty years of landsat science and impacts,” *Remote Sensing of Environment*, vol. 280, p. 113195, 2022.

[32] W. L. Barnes and V. V. Salomonson, “Modis: A global imaging spectroradiometer for the earth observing system,” in *Optical Technologies for Aerospace Sensing: A Critical Review*, vol. 10269. SPIE, 1992, pp. 280–302.

[33] V. V. Salomonson, W. Barnes, and E. J. Masuoka, “Introduction to modis and an overview of associated activities,” *Earth Science Satellite Remote Sensing: Vol. 1: Science and Instruments*, pp. 12–32, 2006.

[34] A. Eckardt, J. Horack, F. Lehmann, D. Krutz, J. Drescher, M. Whorton, and M. Soutullo, “Desis (dlr earth sensing imaging spectrometer for the iss-muses platform),” *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2015-Novem, pp. 1457–1459, 11 2015.

[35] R. O. Green, N. Mahowald, C. Ung, D. R. Thompson, L. Bator, M. Bennet, M. Bernas, N. Blackway, C. Bradley, J. Cha, P. Clark, R. Clark, D. Cloud, E. Diaz, E. Ben Dor, R. Duren, M. Eastwood, B. L. Ehlmann, L. Fuentes, P. Ginoux, J. Gross, Y. He, O. Kalashnikova, W. Kert, D. Keymeulen, M. Klimesh, D. Ku, H. Kwong-Fu, E. Liggett, L. Li, S. Lundeen, M. D. Makowski, A. Mazer, R. Miller, P. Mouroulis, B. Oaida, G. S. Okin, A. Ortega, A. Oyake, H. Nguyen, T. Pace, T. H. Painter, J. Pempejian, C. P. Garcia-Pando, T. Pham, B. Phillips, R. Pollock, R. Purcell, V. Realmuto, J. Schoolcraft, A. Sen, S. Shin, L. Shaw, M. Soriano, G. Swayze, E. Thingvold, A. Vaid, and J. Zan, “The earth surface mineral dust source investigation: An earth science imaging spectroscopy mission,” in *2020 IEEE Aerospace Conference*, 2020, pp. 1–15.

[36] R. O. Green, “The nasa earth venture instrument, earth surface mineral dust source investigation (emit),” in *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 5004–5006.

[37] K. Obata, S. Tsuchida, I. Nagatani, H. Yamamoto, T. Kouyama, Y. Yamada, Y. Yamaguchi, and J. Ishii, “An overview of iss hisui hyperspectral imager radiometric calibration,” in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2016, pp. 1924–1927.

[38] H. Dierssen, M. Gierach, L. Guild, A. Mannino, J. Salisbury, S. Schollaert Uz, J. Scott, P. Townsend, K. Turpie, M. Tzortziou *et al.*, “Synergies between nasa’s hyperspectral aquatic missions pace, glimr, and sbg: Opportunities for new science and applications,” *Journal of Geophysical Research: Biogeosciences*, vol. 128, no. 10, p. e2023JG007574, 2023.

[39] ESA and EU, “S2 mission overview.” [Online]. Available: https://sentiwiki.copernicus.eu/web/s2-mission

[40] C. Program, “Sentinel-2a: Exceptional temporary extension campaign starting in march 2025,” Jan 2025. [Online]. Available: https://sentinels.copernicus.eu/-/sentinel-2a-exceptional-temporary-extension-campaign-starting-in-march-2025

[41] Jun 2024. [Online]. Available: https://sentinels.copernicus.eu/-/copernicus-sentinel-2c-spectral-response-functions

[42] 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, p. 2291, 2020.

[43] “Organization - EnMAP — enmap.org,” https://www.enmap.org/mission/organization/, [Accessed 18-06-2025].

[44] E. O. C. of DLR. [Online]. Available: https://www.enmap.org/science/

[45] [Online]. Available: https://www.enmap.org/data_access/

[46] M. Pato and M. Bachmann, “Level 1b processor (systematic and radiometric correction) atbd,” Nov 2024. [Online]. Available: https://www.enmap.org/data/doc/EN-PCV-TN-4006_Level_1B_Processor_Systematic_and_Radiometric_Correction.pdf

[47] M. Schneider, “Level 1c processor (geometric correction) atbd,” Dec 2024. [Online]. Available: https://www.enmap.org/data/doc/EN-PCV-TN-5006_Level_1C_Processor_Geometric_Correction.pdf

[48] L. Main-Knorn and et al., *Sen2Cor Configuration and User Manual*, 2017, eSA, https://step.esa.int/main/third-party-plugins-2/sen2cor/.

[49] R. Richter and D. Schläpfer, "Atmospheric/topographic correction for satellite imagery (atcor-2/3 user guide, version 9.3. 0, october 2019)," 2019.

[50] D. R. Thompson, B. Gao, and R. O. Green, "Isofit: Imaging spectrometer optimal fit," https://github.com/isofit/isofit, 2020, jet Propulsion Laboratory, California Institute of Technology.

[51] P. Townsend, M. M. Gierach, C. Ade *et al.*, "Sister: Prisma l2a corrected surface reflectance 30 m v001," *ORNL DAAC, Oak Ridge, Tennessee, USA*, 2023.

[52] ESA, "Sentinel-2 Level-2A Algorithm Theoretical Basis Document (ATBD) - S2-PDGS-MPC-ATBD-L2A 2.10," nov 2021.

[53] L. Cenci, M. Galli, C. Santella, V. Boccia, and C. Albinet, "Analyzing the impact of the different instances of the Copernicus DEM dataset on the orthorectification of VHR optical data," in *Proceedings of 2022 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2022, pp. 6001–6004.

[54] Copernicus, "S5p mission overview." [Online]. Available: https://sentiwiki.copernicus.eu/web/s5p-mission

[55] A. Apituley, M. Pedergnana, M. Sneep, J. Pepijn, D. Loyola, J. De Haan, and A. Keppens, "Sentinel-5 precursor/TROPOMI Level 2 Product User Manual Ozone profiles document number : S5P-KNMI-L2-0020-MA."

[56] D. G. Loyola, S. G. García, R. Lutz, A. Argyrouli, F. Romahn, R. J. D. Spurr, M. Pedergnana, A. Doicu, V. M. García, and O. Schüssler, "The operational cloud retrieval algorithms from tropomi on board sentinel-5 precursor," *Atmospheric Measurement Techniques*, vol. 11, pp. 409–427, 1 2018. [Online]. Available: https://amt.copernicus.org/articles/11/409/2018/

[57] National Hurricane Center, "Hurricane otis advisory archive," 2023, accessed: 2025-04-15. [Online]. Available: https://www.nhc.noaa.gov/archive/2023/OTIS.shtml

[58] C. C. Aggarwal, *Neural Networks and Deep Learning: A Textbook.* Springer International Publishing, 2018. [Online]. Available: https://books.google.com.mx/books?id=achqDwAAQBAJ

[59] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control.* John Wiley & Sons, 2015.

[60] I. T. Jollife and J. Cadima, "Principal component analysis: A review and recent developments," 4 2016.

[61] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," *arXiv*, 2 2018. [Online]. Available: http://arxiv.org/abs/1802.03426

[62] D. Peng, Z. Gui, and H. Wu, "Interpreting the curse of dimensionality from distance concentration and manifold effect," *arXiv preprint arXiv:2401.00422*, 2025.

[63] A. Zimek, E. Schubert, and H.-P. Kriegel, "A survey on unsupervised outlier detection in high-dimensional numerical data," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 5, no. 5, pp. 363–387, 2012.

[64] F. Li, P. Zhang, and H. Lu, "Unsupervised band selection of hyperspectral images via multi-dictionary sparse representation," *arXiv preprint arXiv:1802.06983*, 2018.

[65] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, 2012.

[66] C. D. Martin and M. A. Porter, "The extraordinary SVD," *American Mathematical Monthly*, vol. 119, no. 10, pp. 838–851, 12 2012.

[67] C. M. Bishop, *Neural networks for pattern recognition.* Oxford university press, 1995.

[68] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," pp. 279–317, 12 2019.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998.

[70] J. E. Ball, D. T. Anderson, and C. S. Chan, "Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community," *Journal of Applied Remote Sensing*, vol. 11, no. 04, p. 1, sep 2017. [Online]. Available: http://cs-chan.com/source/FADL/Online{_}Paper{_}Summary{_}

[71] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 111–118.

[72] D. Scherer, A. Müller, and S. Behnke, “Evaluation of pooling operations in convolutional architectures for object recognition,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6354 LNCS, no. PART 3. Springer, Berlin, Heidelberg, 2010, pp. 92–101. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-15825-4{_}10

[73] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin, “Generalizing to unseen domains: A survey on domain generalization,” *IEEE Transactions on Knowledge and Data Engineering*, 2021. [Online]. Available: https://arxiv.org/abs/2103.03097

[74] V. N. Vapnik, *Statistical Learning Theory.* New York: Wiley, 1998.

[75] D. Tuia, K. Schindler, B. Demir, X. X. Zhu, M. Kochupillai, S. Džeroski, J. N. van Rijn, H. H. Hoos, F. Del Frate, M. Datcu, V. Markl, B. Le Saux, R. Schneider, and G. Camps-Valls, “Artificial intelligence to advance earth observation: A review of models, recent trends, and pathways forward,” *IEEE Geoscience and Remote Sensing Magazine*, pp. 2–25, 2024.

[76] G. Sumbul and B. Demir, “Label noise robust image representation learning based on supervised variational autoencoders in remote sensing,” in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS).* IEEE, 2023.

[77] T. Burgert, M. Ravanbakhsh, and B. Demir, “On the effects of different types of label noise in multi-label remote sensing image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, 2022.

[78] G. Sumbul and B. Demir, “Generative reasoning integrated label noise robust deep image representation learning,” *IEEE Transactions on Image Processing*, vol. 32, pp. 4529–4542, 2023.

[79] A. K. Aksoy, M. Ravanbakhsh, and B. Demir, “Multi-label noise robust collaborative learning for remote sensing image classification,” *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[80] A. K. Aksoy, M. Ravanbakhsh, T. Kreuziger, and B. Demir, “A consensual collaborative learning method for remote sensing image classification under noisy multi-labels,” in *IEEE International Conference on Image Processing (ICIP).* Anchorage, Alaska, USA: IEEE, 2021, pp. 1314–1318.

[81] European Environment Agency, “Corine land cover—clc 2018 technical guidelines,” European Environment Agency, Tech. Rep. EEA Report No. 07/2018, 2018. [Online]. Available: https://land.copernicus.eu/en/technical-library/clc-2018-technical-guidelines/@@download/file

[82] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, “Bigearthnet: A large-scale benchmark archive for remote sensing image understanding,” *Remote Sensing*, vol. 13, no. 6, p. 1104, 2021.

[83] Z. Li, X. Zhang, and P. Xiao, “Spectral index-driven fcn model training for water extraction from multispectral imagery,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 192, pp. 344–360, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271622002283

[84] S. Skakun, J. Wevers, C. Brockmann, G. Doxani, M. Aleksandrov, M. Batič, D. Frantz, F. Gascon, L. Gómez-Chova, O. Hagolle, D. López-Puigdollers, J. Louis, M. Lubej, G. Mateo-García, J. Osman, D. Peressutti, B. Pflug, J. Puc, R. Richter, J. C. Roger, P. Scaramuzza, E. Vermote, N. Vesel, A. Zupanc, and L. Žust, “Cloud mask intercomparison exercise (cmix): An evaluation of cloud masking algorithms for landsat 8 and sentinel-2,” *Remote Sensing of Environment*, vol. 274, p. 112990, 6 2022.

[85] G. Mateo-García, V. Laparra, D. López-Puigdollers, and L. Gómez-Chova, “Transferring deep learning models for cloud detection between landsat-8 and proba-v,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 160, pp. 1–17, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271619302801

[86] D. López-Puigdollers, G. Mateo-García, and L. Gómez-Chova, “Benchmarking deep learning models for cloud detection in landsat-8 and sentinel-2 images,” *Remote Sensing*, vol. 13, no. 5, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/5/992

[87] S. Foga, P. L. Scaramuzza, S. Guo, Z. Zhu, R. D. Dilley, T. Beckmann, G. L. Schmidt, J. L. Dwyer, M. Joseph Hughes, and B. Laue, “Cloud detection algorithm comparison and validation for operational landsat data products,” *Remote Sensing of Environment*, vol. 194, pp. 379–390, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425717301293

[88] S. Qiu, Z. Zhu, and B. He, “Fmask 4.0: Improved cloud and cloud shadow detection in landsats 4–8 and sentinel-2 imagery,” *Remote Sensing of Environment*, vol. 231, p. 111205, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425719302172

[89] Z. Zhu, S. Wang, and C. E. Woodcock, “Improvement and expansion of the fmask algorithm: cloud, cloud shadow, and snow detection for landsats 4–7, 8, and sentinel 2 images,” *Remote Sensing of Environment*, vol. 159, pp. 269–277, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425714005069

[90] D. Frantz, "Force—landsat + sentinel-2 analysis ready data and beyond," *Remote Sensing*, vol. 11, no. 9, 2019. [Online]. Available: https://www.mdpi.com/2072-4292/11/9/1124

[91] D. Frantz, E. Haß, A. Uhl, J. Stoffels, and J. Hill, "Improvement of the fmask algorithm for sentinel-2 images: Separating clouds from bright surfaces based on parallax effects," *Remote Sensing of Environment*, vol. 215, pp. 471–481, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425718302037

[92] D. Frantz, A. Röder, M. Stellmes, and J. Hill, "An operational radiometric landsat preprocessing framework for large-area time series applications," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 7, p. 3928 – 3943, 2016, cited by: 75. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84960539907&doi=10.1109%2fTGRS.2016.2530856&partnerID=40&md5=8dbf97c027e46854bcfbdc2619e68695

[93] J. Wevers, D. Müller, J. Scholze, G. Kirches, R. Quast, and C. Brockmann, "IdePix for Sentinel-2 MSI Algorithm Theoretical Basis Document," Dec. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5788067

[94] "Improving Cloud Detection with Machine Learning — by Anze Zupanc — Planet Stories — Medium." [Online]. Available: https://medium.com/sentinel-hub/improving-cloud-detection-with-machine-learning-c09dc5d7cf13

[95] "On cloud detection with multi-temporal data — by Jernej Puc — Planet Stories — Medium." [Online]. Available: https://medium.com/sentinel-hub/on-cloud-detection-with-multi-temporal-data-f64f9b8d59e5

[96] S. Skakun, E. F. Vermote, J.-C. Roger, C. O. Justice, and J. G. Masek, "Validation of the lasrc cloud detection algorithm for landsat 8 images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2439–2446, 2019.

[97] S. Skakun, E. F. Vermote, A. E. S. Artigas, W. H. Rountree, and J.-C. Roger, "An experimental sky-image-derived cloud validation dataset for sentinel-2 and landsat 8 satellites over nasa gsfc," *International Journal of Applied Earth Observation and Geoinformation*, vol. 95, p. 102253, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0303243420308965

[98] O. Hagolle, M. Huc, D. V. Pascual, and G. Dedieu, "A multi-temporal method for cloud detection, applied to formosat-2, venµs, landsat and sentinel-2 images," *Remote Sensing of Environment*, vol. 114, no. 8, pp. 1747–1755, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425710000908

[99] O. Hagolle, M. Huc, C. Desjardins, S. Auer, and R. Richter, "Maja algorithm theoretical basis document," Mar. 2018. [Online]. Available: https://doi.org/10.5281/zenodo.1209633

[100] J. Louis, V. Debaecker, B. Pflug, M. Main-Knorn, J. Bieniarz, U. Mueller-Wilm, E. Cadau, and F. Gascon, "Sentinel-2 sen2cor: L2a processor for users," in *Proceedings living planet symposium 2016*. Spacebooks Online, 2016, pp. 1–8.

[101] J. Louis, O. Devignot, and L. Pessiot, "Level-2a algorithm theoretical basis document," *Remote Sensing Systems: Santa Rosa, CA, USA*, vol. 78, 2021.

[102] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, 2022.

[103] X. Yuan, J. Shi, and L. Gu, "A review of deep learning methods for semantic segmentation of remote sensing imagery," *Expert Systems with Applications*, vol. 169, p. 114417, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417420310836

[104] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 9, pp. 5046–5063, 2018.

[105] S. Jozdani, D. Chen, D. Pouliot, and B. Alan Johnson, "A review and meta-analysis of generative adversarial networks and their applications in remote sensing," *International Journal of Applied Earth Observation and Geoinformation*, vol. 108, p. 102734, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0303243422000605

[106] J. Feng, H. Yu, L. Wang, X. Cao, X. Zhang, and L. Jiao, "Classification of hyperspectral images based on multiclass spatial–spectral generative adversarial networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5329–5343, 2019.

[107] A. A. Aleissaee, A. Kumar, R. M. Anwer, S. Khan, H. Cholakkal, G.-S. Xia, and F. S. Khan, "Transformers in remote sensing: A survey," *Remote Sensing*, vol. 15, no. 7, 2023. [Online]. Available: https://www.mdpi.com/2072-4292/15/7/1860

[108] Z. Zhong, Y. Li, L. Ma, J. Li, and W.-S. Zheng, "Spectral–spatial transformer network for hyperspectral image classification: A factorized architecture search framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[109] Y. Bazi, L. Bashmal, M. M. A. Rahhal, R. A. Dayil, and N. A. Ajlan, "Vision transformers for remote sensing image classification," *Remote Sensing*, vol. 13, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/3/516

[110] S. Ghaffarian, J. Valente, M. van der Voort, and B. Tekinerdogan, "Effect of attention mechanism in deep learning-based remote sensing image processing: A systematic literature review," *Remote Sensing*, vol. 13, no. 15, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/15/2965

[111] H. Sun, X. Zheng, X. Lu, and S. Wu, "Spectral–spatial attention network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 5, pp. 3232–3245, 2020.

[112] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, pp. 6232–6251, 10 2016.

[113] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network," *Remote Sensing 2017, Vol. 9, Page 67*, vol. 9, p. 67, 1 2017. [Online]. Available: https://www.mdpi.com/2072-4292/9/1/67/htmhttps://www.mdpi.com/2072-4292/9/1/67

[114] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors*, vol. 2015, pp. 1–12, 2015.

[115] P. Jia, M. Zhang, W. Yu, F. Shen, and Y. Shen, "Convolutional neural network based classification for hyperspectral data," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2016-November, pp. 5075–5078, 11 2016.

[116] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, pp. 234–241, 2015. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-24574-4_28

[117] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised cnn segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1818–1827.

[118] E. Padilla-Zepeda, K. Alonso, R. De Los Reyes, D. Torres-Roman, and A. P. Pertiwi, "A deep learning approach for imagery masking of spectral sensors," in *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, 2024, pp. 8541–8546.

[119] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognition*, vol. 91, pp. 216–231, 7 2019.

[120] D. Chicco, "Ten quick tips for machine learning in computational biology," *BioData mining*, vol. 10, no. 1, p. 35, 2017.

[121] O. Hagolle, "Can reflectances be greater than 1? – séries temporelles," Sep 2019. [Online]. Available: https://www.cesbio.cnrs.fr/multitemp/16885-2/

[122] ESA S2 MSI ESL Team, "Sentinel-2 Annual Performance Report-Year 2023," ESA, Tech. Rep., 2024.

[123] V. Zekoll, M. Main-Knorn, K. Alonso, J. Louis, D. Frantz, R. Richter, and B. Pflug, "Comparison of masking algorithms for sentinel-2 imagery," *Remote Sensing*, vol. 13, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/1/137

[124] D. G. Loyola, S. Gimeno García, R. Lutz, A. Argyrouli, F. Romahn, R. J. D. Spurr, M. Pedergnana, A. Doicu, V. Molina García, and O. Schüssler, "The operational cloud retrieval algorithms from TROPOMI on board Sentinel-5 Precursor," *Atmospheric Measurement Techniques*, vol. 11, no. 1, pp. 409–427, 1 2018. [Online]. Available: https://amt.copernicus.org/articles/11/409/2018/

[125] B. Holben, T. Eck, I. Slutsker, D. Tanré, J. Buis, A. Setzer, E. Vermote, J. Reagan, Y. Kaufman, T. Nakajima, F. Lavenu, I. Jankowiak, and A. Smirnov, "Aeronet—a federated instrument network and data archive for aerosol characterization," *Remote Sensing of Environment*, vol. 66, no. 1, pp. 1 – 16, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0034425798000315

[126] T. Wang, J. Shi, L. Husi, T. Zhao, D. Ji, C. Xiong, and B. Gao, "Effect of solar-cloud-satellite geometry on land surface shortwave radiation derived from remotely sensed data," *Remote Sensing*, vol. 9, no. 7, 2017. [Online]. Available: https://www.mdpi.com/2072-4292/9/7/690

[127] C. J. Hahn, J. London, and R. M. Chervin, "Total cloud cover and cloud type," 1986.

[128] S. Compernolle, A. Argyrouli, R. Lutz, M. Sneep, J. C. Lambert, A. Mari Fjæraa, D. Hubert, A. Keppens, D. Loyola, E. O'Connor, F. Romahn, P. Stammes, T. Verhoelst, and P. Wang, "Validation of the Sentinel-5 Precursor TROPOMI cloud data with Cloudnet, Aura OMI O2-O2, MODIS, and Suomi-NPP VIIRS," *Atmospheric Measurement Techniques*, vol. 14, no. 3, pp. 2451–2476, 3 2021.

[129] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 25, 2012, pp. 1097–1105.

[130] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[131] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[132] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017, pp. 1321–1330.

[133] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and interpreting multiclass loss and accuracy assessment metrics for classifications with class imbalance: Guidance and best practices," *Remote Sensing*, vol. 16, no. 3, 2024. [Online]. Available: https://www.mdpi.com/2072-4292/16/3/533

[134] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017. [Online]. Available: https://arxiv.org/abs/1412.6980

[135] B. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica et Biophysica Acta (BBA) - Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0005279575901099

[136] A. Gonzalez-Ramirez, J. Lopez, D. Torres-Roman, and I. Yañez-Vargas, "Analysis of multi-class classification performance metrics for remote sensing imagery imbalanced datasets," *ECORFAN Journal of Quantitative and Statistical Analysis*, vol. 8, pp. 11–17, 2021. [Online]. Available: www.ecorfan.org/bolivia

[137] J. Gorodkin, "Comparing two k-category assignments by a k-category correlation coefficient," *Computational Biology and Chemistry*, vol. 28, no. 5, pp. 367–374, 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1476927104000799

[138] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: an overview," *arXiv e-prints*, 8 2020. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2020arXiv200805756Ghttp://arxiv.org/abs/2008.05756

[139] E. F. Vermote and S. Kotchenova, "Atmospheric correction for the monitoring of land surfaces," *Journal of Geophysical Research: Atmospheres*, vol. 113, no. D23, 2008.

[140] M. Claverie, E. F. Vermote, B. Franch, and J. G. Masek, "Evaluation of the landsat-5 tm and landsat-7 etm+ surface reflectance products," *Remote Sensing of Environment*, vol. 169, pp. 390–403, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425715301188

[141] J. Ju, D. P. Roy, E. Vermote, J. Masek, and V. Kovalskyy, "Continental-scale validation of modis-based and ledaps landsat etm+ atmospheric correction methods," *Remote Sensing of Environment*, vol. 122, pp. 175–184, 2012, landsat Legacy Special Issue. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S003442571200051X

[142] G. Doxani, E. Vermote, J. C. Roger, F. Gascon, S. Adriaensen, D. Frantz, O. Hagolle, A. Hollstein, G. Kirches, F. Li, J. Louis, A. Mangin, N. Pahlevan, B. Pflug, and Q. Vanhellemont, "Atmospheric Correction Inter-Comparison Exercise," *Remote Sensing 2018, Vol. 10, Page 352*, vol. 10, no. 2, p. 352, 2 2018. [Online]. Available: https://www.mdpi.com/2072-4292/10/2/352/htmhttps://www.mdpi.com/2072-4292/10/2/352

[143] G. Doxani, E. F. Vermote, J.-C. Roger, S. Skakun, F. Gascon, A. Collison, L. De Keukelaere, C. Desjardins, D. Frantz, O. Hagolle, M. Kim, J. Louis, F. Pacifici, B. Pflug, H. Poilvé, D. Ramon, R. Richter, and F. Yin, "Atmospheric correction inter-comparison exercise, acix-ii land: An assessment of atmospheric correction processors for landsat 8 and sentinel-2 over land," *Remote Sensing of Environment*, vol. 285, p. 113412, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425722005181

[144] C. B. Lee, D. Traganos, and P. Reinartz, "A simple cloud-native spectral transformation method to disentangle optically shallow and deep waters in sentinel-2 images," *Remote Sensing*, vol. 14, no. 3, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/3/590

[145] MPC, "Data quality report ref. s2-pdgs-mpc-dqr s2 mpc," 3 2018.

[146] D. Loyola, R. Lutz, A. Argyrouli, and R. Spurr, "S5P/TROPOMI ATBD Cloud Products," Jun 2021.

[147] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[148] A. Höhl, I. Obadic, M.-Á. Fernández-Torres, H. Najjar, D. A. B. Oliveira, Z. Akata, A. Dengel, and X. X. Zhu, "Opening the black box: A systematic review on explainable artificial intelligence in remote sensing," *IEEE Geoscience and Remote Sensing Magazine*, vol. 12, no. 4, pp. 261–304, 2024.

[149] S. O. Atik and M. E. Atik, “Optimal band selection using explainable artificial intelligence for machine learning-based hyperspectral image classification,” *Journal of Applied Remote Sensing*, vol. 18, no. 4, p. 042604, 2024. [Online]. Available: https://doi.org/10.1117/1.JRS.18.042604

[150] D. E. Turan, E. Aptoula, A. Ertürk, and G. Taskin, “Interpreting hyperspectral remote sensing image classification methods via explainable artificial intelligence,” in *IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*, 2023, pp. 5950–5953.

[151] A. H. Oveis, E. Giusti, G. Meucci, S. Ghio, and M. Martorella, “Explainability in hyperspectral image classification: A study of xai through the shap algorithm,” in *2023 13th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2023, pp. 1–5.

[152] G. De Lucia, M. Lapegna, and D. Romano, “Towards explainable ai for hyperspectral image classification in edge computing environments,” *Computers and Electrical Engineering*, vol. 103, p. 108381, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790622005985

[153] M. H. Dehghan-Shoar, G. Kereszturi, R. R. Pullanagari, A. A. Orsi, I. J. Yule, and J. Hanly, “A physically informed multi-scale deep neural network for estimating foliar nitrogen concentration in vegetation,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 130, p. 103917, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569843224002711

[154] B. Li, Q. Hu, M. Gao, T. Liu, C. Zhang, and C. Liu, “Physical informed neural network improving the wrf-chem results of air pollution using satellite-based remote sensing data,” *Atmospheric Environment*, vol. 311, p. 120031, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1352231023004570

[155] G. Joseph, *Building earth observation cameras.* CRC press, 2015.

[156] “Dlr - hyspex-sensor in oberpfaffenhofen.” [Online]. Available: https://www.dlr.de/de/forschung-und-transfer/forschungsinfrastruktur/grossforschungsanlagen/hyspex

[157] Q. Zhao, L. Yu, Z. Du, D. Peng, P. Hao, Y. Zhang, and P. Gong, “An overview of the applications of earth observation satellite data: Impacts and future trends,” *Remote Sensing*, vol. 14, no. 8, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/8/1863

[158] Jan 2012. [Online]. Available: https://www.lakesuperior.com/the-lake/natural-world/how-to-identify-cloud-types/

[159] J. R. Schott, *Remote sensing: the image chain approach.* Oxford University Press, 2007.

[160] Wikipedia contributors, "Steradian — Wikipedia, the free encyclopedia," 2025, [Online; accessed 14-January-2025]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Steradian&oldid=1267754343

[161] O. M. Coddington, E. C. Richard, D. Harber, P. Pilewskie, T. N. Woods, K. Chance, X. Liu, and K. Sun, "The tsis-1 hybrid solar reference spectrum," *Geophysical Research Letters*, vol. 48, no. 12, p. e2020GL091709, 2021, e2020GL091709 2020GL091709. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020GL091709

[162] J. M. Fontenla, J. Harder, W. Livingston, M. Snow, and T. Woods, "High-resolution solar spectral irradiance from extreme ultraviolet to far infrared," *Journal of Geophysical Research: Atmospheres*, vol. 116, no. D20, 2011. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011JD016032

[163] G. Thuillier, M. Hersé, D. Labs, T. Foujols, W. Peetermans, D. Gillotay, P. Simon, and H. Mandel, "The solar spectral irradiance from 200 to 2400 nm as measured by the solspec spectrometer from the atlas and eureca missions," *Solar Physics*, vol. 214, no. 1, May 2003. [Online]. Available: https://doi.org/10.1023/A:1024048429145

[164] J. Zhang and J. Li, *Spatial Cognitive Engine Technology.* Elsevier, 2023.

[165] "Esa - types of orbits." [Online]. Available: https://www.esa.int/Enabling_Support/Space_Transportation/Types_of_orbits

[166] "Copernicus programme levels up with sentinel-2c in orbit." [Online]. Available: https://defence-industry-space.ec.europa.eu/copernicus-programme-levels-sentinel-2c-orbit-2024-09-05_en

[167] "What is enmap capable of?" [Online]. Available: https://www.ohb.de/en/magazine/what-is-enmap-capable-of

[168] J. Jia, Y. Wang, J. Chen, R. Guo, R. Shu, and J. Wang, "Status and application of advanced airborne hyperspectral imaging technology: A review," *Infrared Physics & Technology*, vol. 104, p. 103115, 1 2020.

[169] A. Baumgartner, P. Gege, C. Köhler, K. Lenhard, and T. Schwarzmaier, "Characterisation methods for the hyperspectral sensor HySpex at DLR's calibration home base," in *Sensors, Systems, and Next-Generation Satellites XVI*, R. Meynart, S. P. Neeck, and H. Shimoda, Eds., vol. 8533, International

Society for Optics and Photonics. SPIE, 2012, p. 85331H. [Online]. Available: https://doi.org/10.1117/12.974664

[170] S. Bourennane, C. Fossati, and T. Lin, "Noise Removal Based on Tensor Modelling for Hyperspectral Image Classification," *Remote Sensing*, vol. 10, no. 9, p. 1330, aug 2018. [Online]. Available: http://www.mdpi.com/2072-4292/10/9/1330

[171] L. Alparone, M. Selva, B. Aiazzi, S. Baronti, F. Butera, and L. Chiarantini, "Signal-dependent noise modelling and estimation of new-generation imaging spectrometers," in *WHISPERS '09 - 1st Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2009.

[172] B. Rasti, P. Scheunders, P. Ghamisi, G. Licciardi, and J. Chanussot, "Noise Reduction in Hyperspectral Imagery: Overview and Application," *Remote Sensing*, vol. 10, no. 3, p. 482, mar 2018. [Online]. Available: http://www.mdpi.com/2072-4292/10/3/482

[173] H. Faraji and W. J. MacLean, "CCD noise removal in digital images," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2676–2685, sep 2006.

[174] A. K. Jain, *Fundamentals of Digital Image Processing.* USA: Prentice-Hall, Inc., 1989.

[175] X. Liu, S. Bourennane, and C. Fossati, "Reduction of signal-dependent noise from hyperspectral images for target detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 9, pp. 5396–5411, 2014.

[176] J. Rouse, R. Haas, J. Schell, and D. Deering, "Monitoring vegetation systems in the great plains with erts," in *Third Earth Resources Technology Satellite-1 Symposium*, vol. 1. NASA SP-351, 1973.

[177] S. McFeeters, "The use of the normalized difference water index (ndwi) in the delineation of open water features," *International Journal of Remote Sensing*, vol. 17, no. 7, pp. 1425–1432, 1996.

[178] B.-C. Gao, "Ndwi—a normalized difference water index for remote sensing of vegetation liquid water from space," *Remote Sensing of Environment*, vol. 58, no. 3, pp. 257–266, 1996.

[179] Y. Zha, J. Gao, and S. Ni, "Use of normalized difference built-up index in automatically mapping urban areas from tm imagery," *International Journal of Remote Sensing*, vol. 24, no. 3, pp. 583–594, 2003.

[180] A. Huete, "A soil-adjusted vegetation index (savi)," *Remote Sensing of Environment*, vol. 25, no. 3, pp. 295–309, 1988.

[181] A. Huete, K. Didan, T. Miura, E. Rodriguez, X. Gao, and L. Ferreira, “Overview of the radiometric and biophysical performance of the modis vegetation indices,” *Remote Sensing of Environment*, vol. 83, no. 1-2, pp. 195–213, 2002.

[182] J. Qi, A. Chehbouni, A. Huete, Y. Kerr, and S. Sorooshian, “A modified soil adjusted vegetation index,” *Remote Sensing of Environment*, vol. 48, no. 2, pp. 119–126, 1994.

[183] M. Martín, I. Gómez, and E. Chuvieco, “Burned area mapping using logistic regression and decision tree algorithms with modis data,” in *Proceedings of the 5th International Workshop on Remote Sensing and GIS Applications to Forest Fire Management*, 2005.

[184] D. Hall, G. Riggs, and V. Salomonson, “Development of methods for mapping global snow cover using moderate resolution imaging spectroradiometer data,” *Remote Sensing of Environment*, vol. 54, no. 2, pp. 127–140, 1995.

[185] I. of the day, “The water quality of lake chapala in méxico, threatened by an algae bloom,” Feb 2024. [Online]. Available: https://www.copernicus.eu/en/media/image-day-gallery/water-quality-lake-chapala-mexico-threatened-algae-bloom

[186] NASA Science Editorial Team, “Earth’s atmosphere: A multi-layered cake,” October 2019, accessed: 2025-04-05. [Online]. Available: https://science.nasa.gov/earth/earth-atmosphere/earths-atmosphere-a-multi-layered-cake/

[187] B.-C. Gao and A. F. Goetz, “Cirrus cloud detection from airborne imaging spectrometer data using the 1.38 μm water vapor band,” *Journal of Geophysical Research: Atmospheres*, vol. 98, no. D2, pp. 16 721–16 729, 1993.

[188] M. D. King, C. L. Parkinson, K. C. Partington, and R. P. Williams, *Remote sensing of the troposphere.* NASA Reference Publication 1298, 1992.

[189] K.-N. Liou, *An Introduction to Atmospheric Radiation*, 2nd ed. Academic Press, 2002.

[190] S. Platnick, P. A. Hubanks, M. D. King, K. G. Meyer, and G. Wind, “Modis atmosphere l2 cloud product (06_l2) algorithm theoretical basis document,” NASA Goddard Space Flight Center, Tech. Rep., 2017, version 6.1, available at https://modis-atmos.gsfc.nasa.gov.

[191] J. Martins, L. Remer, and Y. Kaufman, “Comparison of the modis aerosol retrievals on the terra and aqua platforms,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 5, pp. 1352–1363, 2002.

[192] C. J. Stubenrauch, W. B. Rossow, S. Kinne, C. Jakob, D. E. Waliser, and C. Hahn, “Assessment of global cloud datasets from satellites: Project and database initiated by the gewex radiation panel,” *Bulletin of the American Meteorological Society*, vol. 94, no. 7, pp. 1031–1049, 2013.

List of Figures

List of Tables

This page is intentionally left blank.

Appendices

This page is intentionally left blank.

Appendix A

Background Support

A.1 Optical Remote Sensing

A.1.1 Physics of Light

"A photon is a discrete particle of EM energy having no mass, no electric charge, and an indefinite life" [18]. "It is generally easier to think of energy as being transferred in terms of energy packets or *quanta* in accordance with quantum theory"[159]. The *energy* (q) [$\mathrm{J} = \mathrm{kg} \cdot \mathrm{m}^2 \cdot \mathrm{s}^{-2}$] of any photon is related to its frequency as follows, where $h = 6.626\,068 \times 10^{-34}\,\mathrm{kg\,m^2\,s^{-1}}$ is the *Planck's constant* and v [s^{-1}] is the *frequency*, as shown in equation A.1.

$$q = hv \ [\mathrm{J}] \tag{A.1}$$

Frequency is defined as follows [159]:

$$v = \frac{c}{\lambda} \ \left[\mathrm{s}^{-1}\right] \tag{A.2}$$

where c [$\mathrm{m\,s^{-1}}$] is the *speed of light* and λ [µm] the *wavelength*.
The *total energy* (Q) in a beam or ray is the sum of q over all frequencies [159]:

$$Q = \sum_i q_i = \sum_i n_i h v_i, \tag{A.3}$$

where n_i is the number of photons at frequency i.

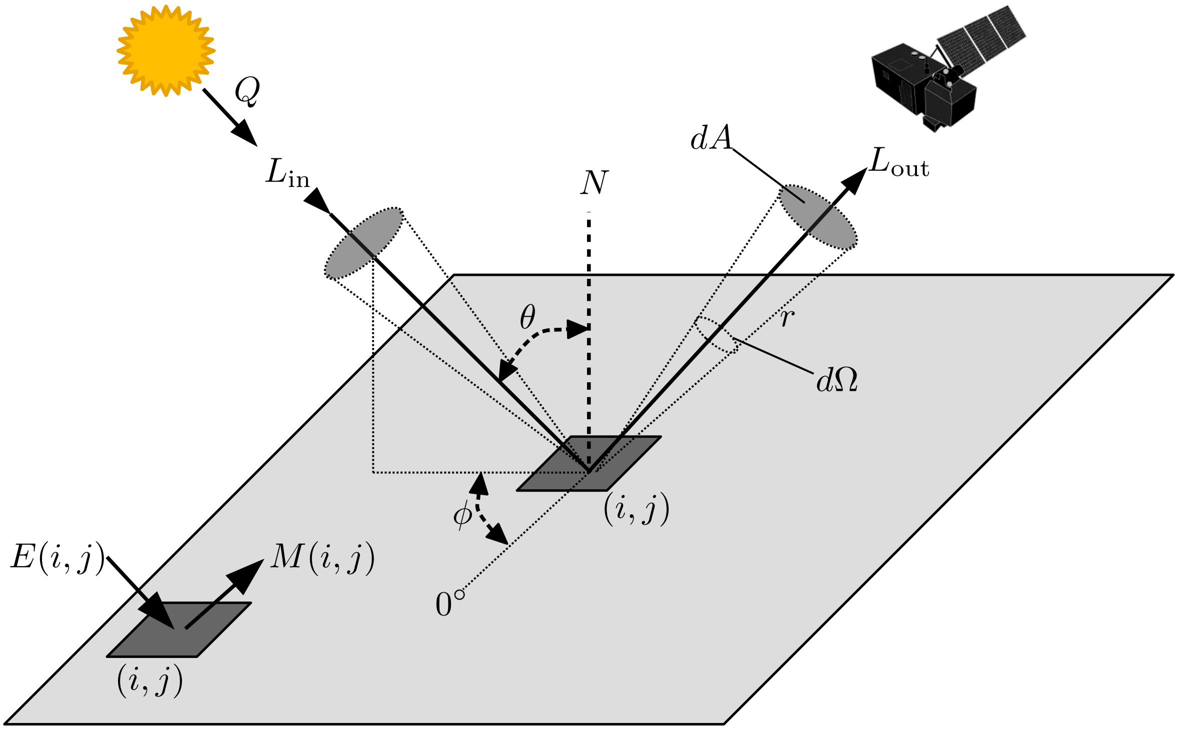


Figure A.1: Illustration of radiometry functions.

A.1.1.1 Radiometric Measures

It is convenient to express the beam in terms of the rate at which the energy is propagating, and not in terms of total energy as in equation A.3. Therefore, the rate of flow of energy is called the *radiant flux* (Φ), defined as the derivative of Q with respect to time [159]:

$$\Phi = \frac{dQ}{dt} \; [\mathrm{W}]. \tag{A.4}$$

To measure the rate at which the radiant flux is delivered to a surface, the term *irradiance* (E) is defined as [159]:

$$E(i,j) = \frac{d\Phi}{dA} \; \left[\mathrm{W\,m^{-2}}\right], \tag{A.5}$$

where dA [m^2] is an area element on the surface defined by (i, j), generic spatial location parameters. In the same way, a very similar term is *radiant exitance*, defined as follows:

$$M(i,j) = \frac{d\Phi}{dA} \; \left[\mathrm{W\,m^{-2}}\right], \tag{A.6}$$

which describes the same concept of irradiance, but instead of the radiant flux delivered to a surface, it is the radiant flux radiated or reflected from a surface.

Both concepts, irradiance and exitance, are defined in terms of the area, but do not consider angular or directional information. To address this, the *radiant intensity* is defined as follows [159]:

$$I(\theta, \phi) = \frac{d\Phi}{d\Omega} \left[\mathrm{W\,sr^{-1}}\right], \tag{A.7}$$

where θ and ϕ are generic orientation angles. $d\Omega$ [sr] is the element of the solid angle. *Steradian*, also named *square radian*, is the unit of solid angle for three-dimensional geometry, representing a right circular cone projected onto a sphere, defining a spherical cap, analogous to the radian for planar angles [160], defined as:

$$\Omega = \frac{A}{r^2} \,[\mathrm{sr}], \tag{A.8}$$

where the area (A) is the surface area of the spherical cap and r the radius of the sphere.

Lastly, the term *radiance* (L) is the most complex of all of these terms, but also the most used for defining the received flux at a sensor, defined as [159]:

$$L(i, j, \theta, \phi) = \frac{d^2\Phi}{dA \cos\theta d\Omega} \left[\mathrm{W\,m^{-2}\,sr^{-1}}\right], \tag{A.9}$$

where i and j define the generic location in the plane, and θ and ϕ define the direction relative to the normal (N) to the plane. Note that I and L are defined for incoming and outgoing radiant flux.

These concepts were briefly described to provide an intuitive understanding of radiometric measurements, but they depend on additional factors, such as the surface that interacts with the radiant flux. However, these short definitions are sufficient to address the needs of the presented work.

A.1.1.2 Transmission, Reflection and Absorption

Through their interaction with the medium, in particular with the Earth's atmosphere and surface, photons can be transmitted, reflected or absorbed [18], and depending on the composition of the matter with which it interacts, these effects happen in different proportions in different wavelengths.

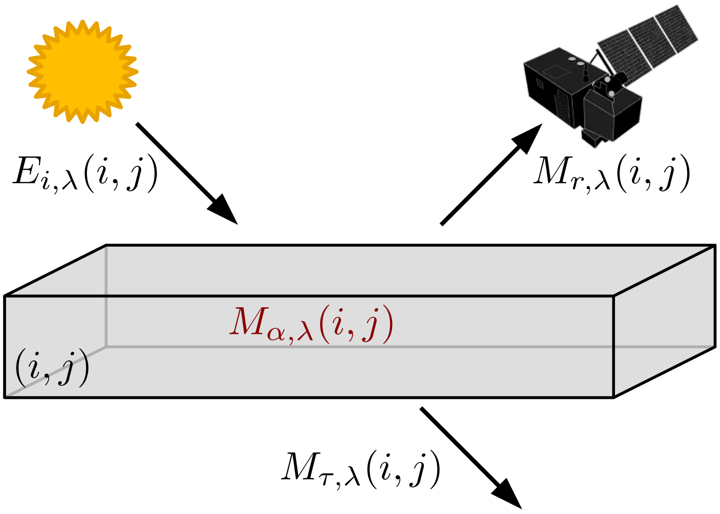


Figure A.2: Illustrated transmission, reflection and absorption.

Considering an object with two sides, illuminating one side, with the ability to absorb and transform energy into another kind, as shown in Figure A.2, the *transmittance* (τ) is a unitless ratio of the radiant exitance from the darken side (M_τ) to the irradiance (E_i), for each *wavelength* (λ) and generic spatial location (i,j) [159]:

$$\tau_\lambda(i,j) = \frac{M_{\tau,\lambda}(i,j)}{E_{i,\lambda}(i,j)}. \tag{A.10}$$

The *reflectance* (r) is the ratio of the radiant exitance (M_r) to the irradiance, both on the illuminated side [159]:

$$r_\lambda(i,j) = \frac{M_{r,\lambda}(i,j)}{E_{i,\lambda}(i,j)}. \tag{A.11}$$

Lastly, the *absorptance* (α) is the ratio between the exitance (M_α), converted in another kind of energy (i.e. heat), to the irradiance [159]:

$$\alpha_\lambda(i,j) = \frac{M_{\alpha,\lambda}(i,j)}{E_{i,\lambda}(i,j)}. \tag{A.12}$$

"Since conservation of energy requires all the incident flux to be absorbed, transmitted, or reflected" [159], such that:

$$\tau_\lambda(i,j) + r_\lambda(i,j) + \alpha_\lambda(i,j) = 1, \tag{A.13}$$

for opaque objects, which is the case of most solids, where $\tau_\lambda(i,j) = 0$ [18, 159], and solved for the reflectance:

$$r_\lambda(i,j) = 1 - \alpha_\lambda(i,j), \tag{A.14}$$

reflectivity is also commonly referred to as *albedo*.

The direction of flux reflection depends on the incident angle (relative to the normal angle N) and the roughness of the reflecting surface. In the case of a smooth surface, reflection is *specular*, which is analogous to a mirror reflecting light. In the case of a rough surface, reflection is *diffuse* (also named *Lambertian*) scattering the flux in different directions and proportions. This scattering behaviors are illustrated in Figure A.3.

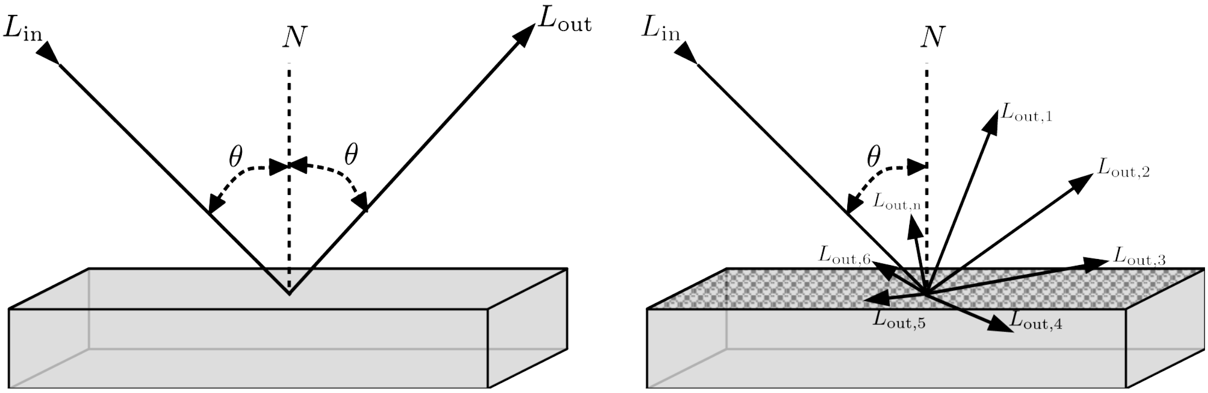


Figure A.3: Left: Specular reflection. Right: Diffuse reflection.

In summary, and in general terms, the reflection of radiant flux depends on the absorptance and transmittance of the flux for each wavelength, which is associated with the matter, the incidence angles, the texture, and other factors not mentioned yet. From the perspective of optical remote sensing, the only measurable quantity is the reflected flux, captured by the sensor, while the rest can be retrieved from physical models, such as by computing the radiance from the capturing angles, which are determined by the position of the sun and the capturing spaceborne camera.

A.1.1.3 Solar Radiation and Atmospheric Transmittance

The energy emitted by the sun is the main source of illumination for optical remote sensing of the Earth's surface, but the energy is not emitted uniformly across all wavelengths. The spectrum received outside the atmosphere is called exoatmospheric spectral irradiance (or extraterrestrial solar irradiance at 1 Astronomical Unit (AU) from the sun), and it varies over time depending on solar activity. Solar irradiance is the primary source of energy reaching the Earth, and its measurement has been conducted from satellites and correlated with the observation of sunspot activity [18]. There are different solar radiation models [161, 162, 163], which are constantly updated to accurately estimate irradiance from measured reflectance. Figure A.4 shows an example of the measured solar irradiance spectrum.

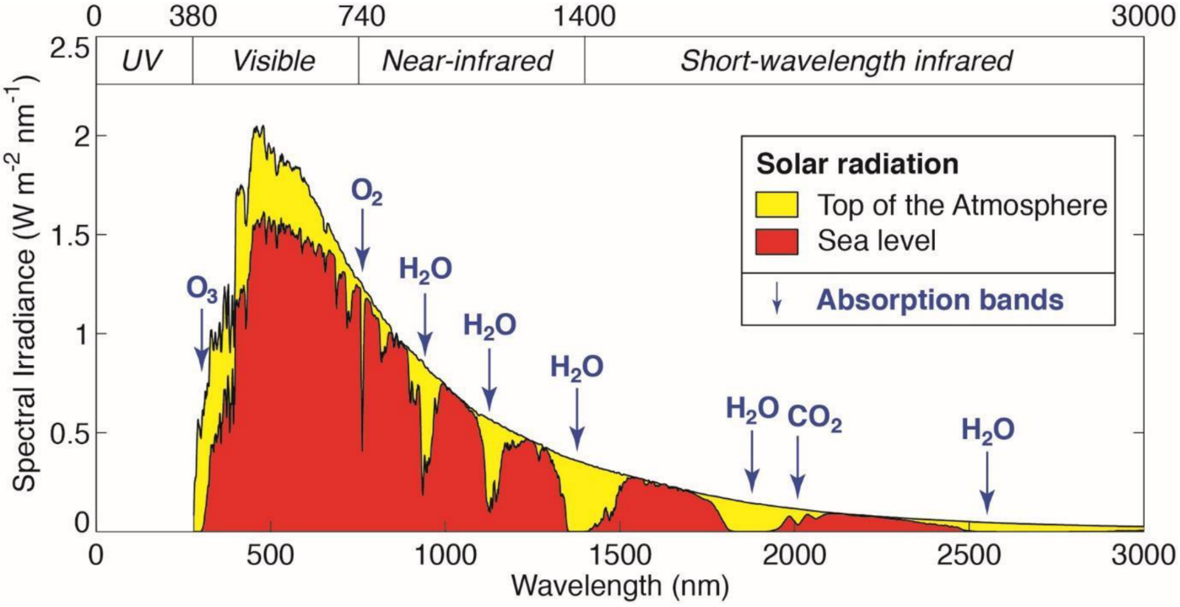


Figure A.4: Solar radiation spectrum above atmosphere and at surface (sea level) with the atmospheric absorption bands with the associated molecules. Data derived from the ASTM. Source: EnMAP science plan [26].

In order to characterize the received radiant flux at the Earth's surface, in addition to the incoming solar radiation, another factor to consider are the interactions of constituents of atmosphere with the solar spectrum.

Water vapor, ozone, and various atmospheric gases interact with radiant flux primarily through absorption and transmission, while scattering processes account for redirection. The extent of these interactions depends on the gases' composition and particle size. As a result, the atmosphere significantly attenuates specific wavelengths (known as absorption bands) which are associated with abundant components such as O_3, O_2, Water (H_2O) in the form of WV, and Carbon Dioxide (CO_2), among others.

The wavelengths ranges that pass through the atmosphere with minimal absorption are referred to as atmospheric windows. These are the portions of the EM spectrum transmitted between absorption bands. However, their exact characteristics depend on atmospheric conditions at a given time and location, as molecular concentrations continuously fluctuate. Thus, the radiant flux reaching the Earth's surface is the one transmitted in the atmospheric windows at the specific capture time. An example of atmospheric transmittance, specifying absorption bands is shown in Figure A.4.

A.1.2 Optical Sensors

A.1.2.1 Operating Principles

An *electromagnetic sensor* takes the incoming energy reflected by the observed object and transforms it into a voltage through a combination of input electrical power and sensor material that is responsive to the type of energy being detected. The output voltage waveform is the response of the sensor, and a digital quantity is obtained by *digitizing* that response [27].

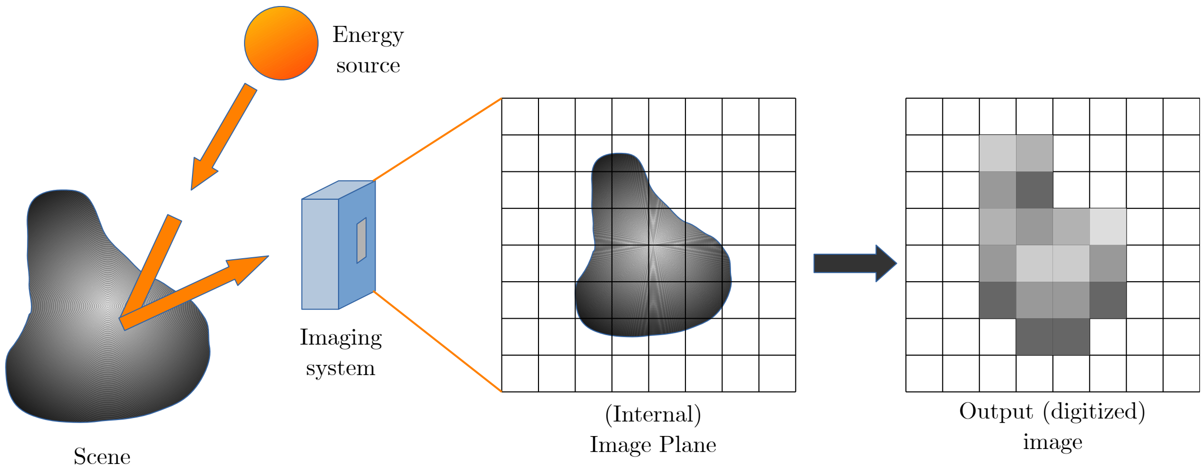


Figure A.5: Digital image acquisition.

"An image may be defined as a two-dimensional function $f(x, y)$ where x and y are *spatial* (plane) coordinates, and the amplitude of f at any pair of coordinated (x, y) is called the *intensity*. When x,y and the intensity values of f are all finite, discrete quantities, thus, the image is a *digital image*" [27].

Digital images are captured by sensors that measure photon flux, and these consist of a Charge-Coupled Device (CCD) together with a lens and a field stop for controlling the amount of light striking the sensor. Using an analogy, the CCD, made of silicon pixels sensitive to photons, functions like silver halide-coated film in traditional cameras [18].

"An image may be continuous with respect to the x and y coordinates and also in amplitude. To digitize it, the function is sampled in both coordinates and also in amplitude. Digitizing the coordinate values is called *sampling*. Digitizing the amplitude is called *quantization*" [27]. An example of a scene, the sensor measurement, sampling, quantization, and the final result are shown in Figure A.6.

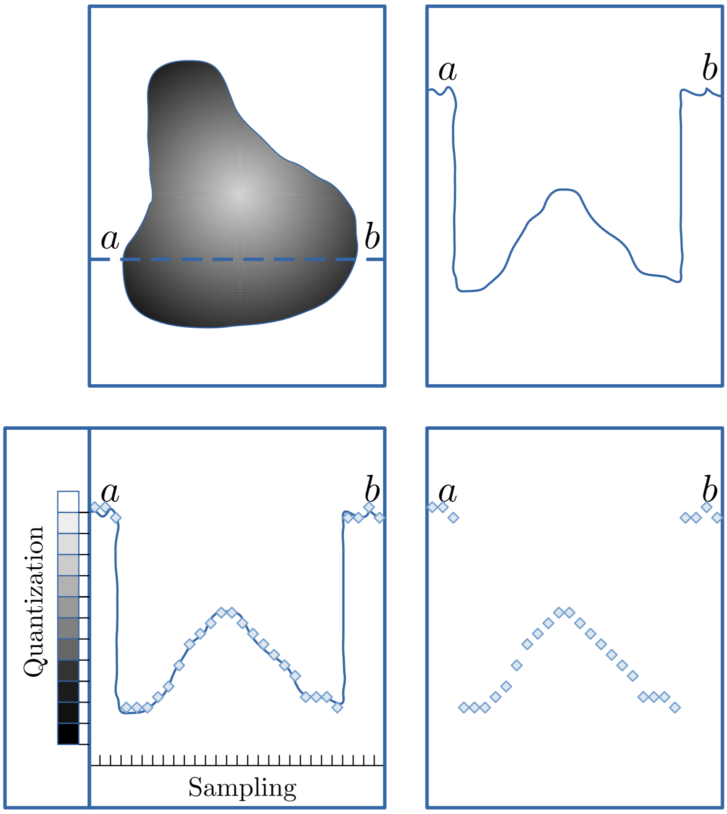


Figure A.6: Sampling and quantization from continuous to digital image.

A.1.2.2 Spectral Response and Spectral Resolution

Given the discrete nature of digital images, the focus is on sampling the EM spectrum of the received flux at the sensor. To achieve this, the analysis centers on the *spectral density*, which is expressed as flux per unit wavelength interval.

Given that, the *responsivity* is described as a bandpass filter in terms of the wavelength (λ), defined as the signal output (S) per unit flux incident on the sensor at the specified wavelength [159]. Thus, the SRF, denoted as R, is defined as follows:

$$R(\lambda) = \frac{dS}{d\Phi(\lambda)} \left[\mathrm{V\,W^{-1}}\right] \tag{A.15}$$

Therefore, the output signal is obtained by integrating the previous expression, which consists of the spectral flux $\Phi(\lambda)$ weighted by the SRF [159]:

$$S = \int_0^\infty \Phi_\lambda R(\lambda) d\lambda \,[\mathrm{V}] \tag{A.16}$$

where the integral is computed only over the interval where the spectral response is nonzero.

Finally, in imaging spectroscopy, it is always important to represent the output signal in radiometric terms. In this case, the photon flux is replaced by reflectance, radiation area, and incident angles:

$$S = \int_0^\infty L_\lambda R(\lambda) A\Omega \cos(\theta) d\lambda \,[\mathrm{V}] \tag{A.17}$$

A common assumption for defining the spectral response is that it has a Gaussian nature (idealized) [18], obtained by estimating the shape based on well-characterized references for calibration (i.e., internal reference lamps or the sensor pointing at the moon). The spectral response is then defined by fitting a Gaussian curve using its central wavelength and the FWHM, also denoted by ($\Delta\lambda$), which corresponds to 50% of the peak height, describing the channel bandwidth. An illustration of a SRF is shown in Figure A.7. Usually, each band has a different spectral response with a different central wavelength and FWHM.

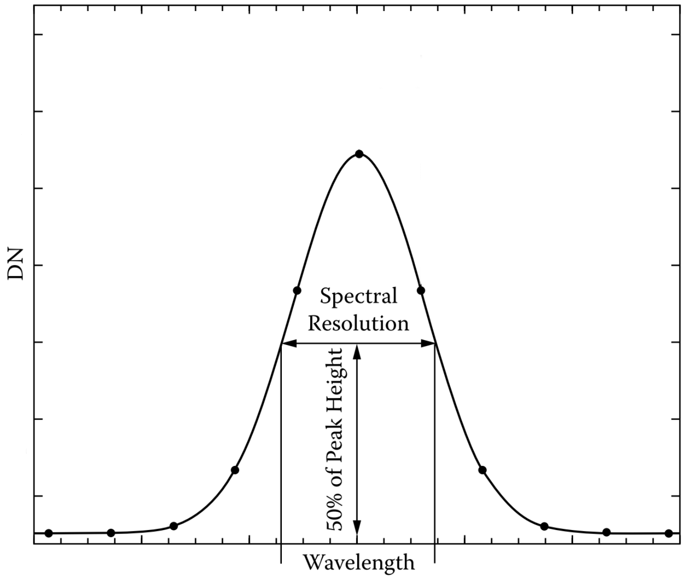


Figure A.7: Spectral response parameters. Image credit: Borengasser [18].

Using the aforementioned concepts, *spectral resolution* refers to spectral features such as the wavelength detection range, number of bands, and FWHM. "The higher the spectral resolution, the narrower the spectral bandwidth required" [164].

Although the ideal SRF is assumed to be Gaussian, its shape varies in practice due to variations in sensor construction and quality. This can be analyzed in pre-flight studies. In Figure A.8, examples of actual SRFs are shown.

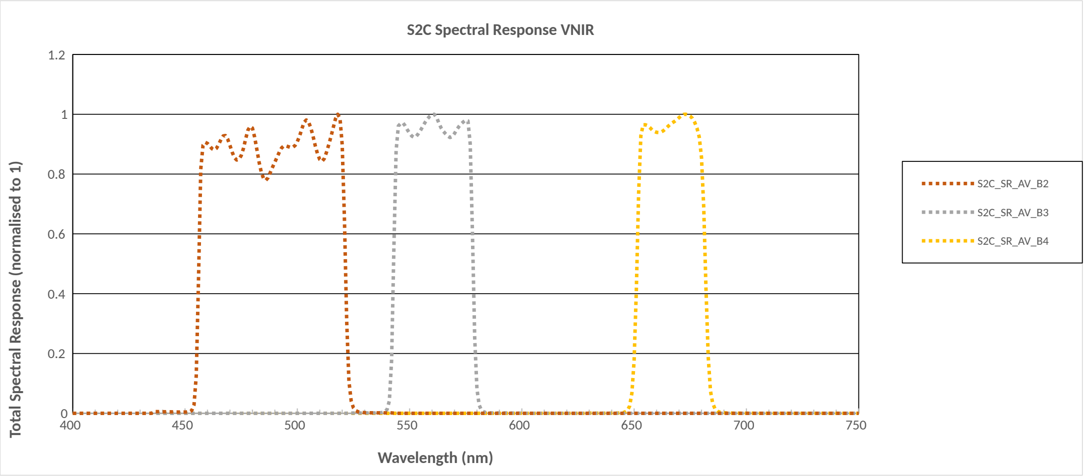


Figure A.8: Spectral response functions of the B4, B3, and B2 bands from the multispectral sensor onboard the Sentinel-2C spacecraft. Credit: Copernicus Program [41].

A.1.2.3 Radiometric Resolution

The voltage generated by the sensor in response to the incoming flux (for each band) is a continuous signal. In order to transmit and store the image, the signal is digitized by an Analog-Digital Converter (ADC), assigning a DN to each ground resolution cell. This process is also known as *quantization*. The range of values that a DN takes defines the *radiometric resolution*, usually depending on the number of bits (b) used to store the intensity of each ground resolution cell, as a *datum* (a sequence of bits) representing a DN as an unsigned integer of b-bits, as follows:

$$\mathrm{DN} \in \mathbb{Z}, \tag{A.18}$$

$$0 \leq \mathrm{DN} \leq 2^b - 1. \tag{A.19}$$

The radiometric resolution is often defined in terms of b-bits. An example of the effect of different radiometric resolutions is shown in Figure A.9.

Figure A.9: Examples of different radiometric resolutions over Chicago, Illinois. Left image of 16-bit ($0 \leq \mathrm{DN} \leq 65,365$). Center image of 8-bit ($0 \leq \mathrm{DN} \leq 255$). Right image of 4-bit ($0 \leq \mathrm{DN} \leq 15$). Image credit: USGS, Public Domain.

There is a trade-off between numerical accuracy and the number of bits used to store each pixel. As seen in Figure A.9, low radiometric resolution loses information about the reflectivity of a ground resolution cell, making it harder to identify fine changes. These errors are known as *quantization noise*, and this will be reviewed in more detail later.

A.1.2.4 Spatial Resolution

Similar to the concept of spectral resolution, the *spatial resolution* describe how distanced are independent measurements on the Earth's surface. To introduce this concept, the first step is to understand intuitively how images are captured.

Most of the currently implemented optical EO missions are based on a push-broom scanner, also known as along-track scanner, capturing a row of pixels per scan as the vehicle moves. The scanning line is perpendicular to the satellite ground track.

The covered ground distance captured by the sensor is known as *ground swath* (also called swath width), and this is defined by the Field Of View (FOV) of the sensor, and the height (H) of the vehicle. These parameters are illustrated in Figure A.10. The resulting captured columns in ground track direction are named *footprints*.

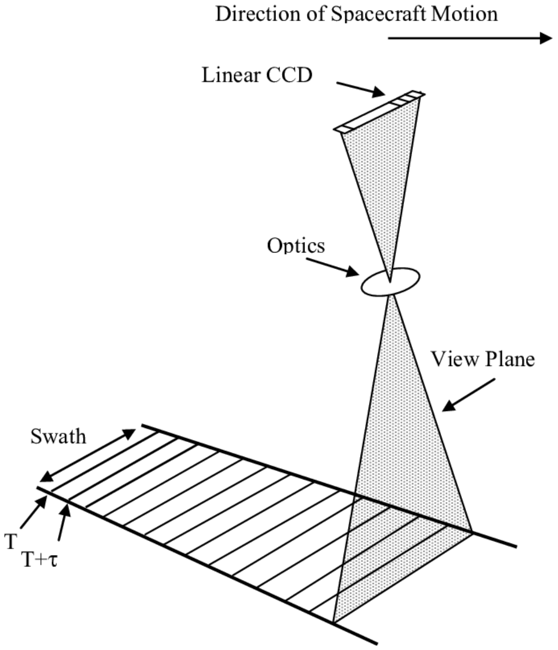


Figure A.10: Push-broom sensor operation by time. Image Credit: [155].

Lastly, to obtain the distance sampled for each pixel, the *ground sampling distance* is computed by dividing the ground swath by the number of captured pixels. Note that this distance is perpendicular to the ground track direction.

To create square pixels (which can also be rectangular), the sensor captures the photon flux for a given time (τ), which depends on the velocity (v) of the vehicle, as illustrated in Figure A.10.

Most monitoring missions capture long strips while traveling forward in a straight line, which are then cropped into individual tiles (usually square) for easy distribution.

A.1.2.5 Sensor Platforms

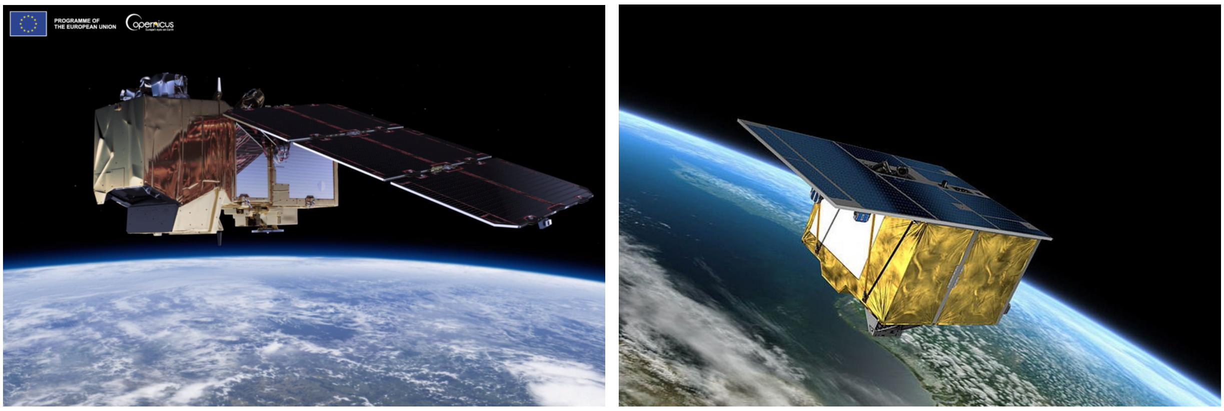
Sensors can be mounted on a wide variety of vehicles, affecting different characteristics of the retrieved images, such as ground sampling distance and ground swath. Given that this work focuses on optical remote sensing missions with spaceborne sensors, the principal characteristics are described below [165]. In addition, other platforms are briefly described.

- **Spaceborne Sensors**: These are spaceborne vehicles (satellites) orbiting the Earth along a specified trajectory. The role of rocket companies is to use launch vehicles carrying payloads (satellites) to pass through the atmosphere and then provide the horizontal velocity needed to enter orbit [165]. In contrast to single-satellite missions, a *constellation* of satellites consists of a network of n satellites, where $n \geq 2$, covering different locations on Earth's surface simultaneously.

  There are different types of orbits that allow remote sensing missions to perform different tasks. The most common orbits for EO missions are described below:

  - Geostationary Orbit (GEO): Satellites flying above the equator at an altitude of 35 786 km in the direction of Earth's rotation (from west to east) matching the velocity, appearing to be static in the sky from the Earth's perspective. This orbit allows pointing to a fixed spot for constant monitoring [165].
  - Low Earth Orbit (LEO): This orbit is relatively low, just high enough (180 km) to avoid effects of atmosphere over the spacecraft (reducing its speed), and the upper limit (2000 km) is defined by the Van Allen belts that could harsh satellites. These satellites could have orbital planes at tilted angles. The low altitude allows capturing images with relatively high resolution [165]. The International Space Station (ISS) is in this orbit at an altitude of 408 km.
  - Polar Orbit (PO): This is a type of LEO, with altitudes ranging from 200 km to 1000 km. These satellites travel from one pole to the other with a small deviation of less than 10°. This orbit is particularly useful for Earth monitoring missions, as the sensor points downward while traveling from pole to pole, while the planet rotates below, allowing coverage of every location on Earth's surface [165].
  - SSO: A particular type of PO in which satellites are synchronized with the Sun, maintaining the same position relative to it. This allows them to capture the same location under consistent illumination conditions (sun-zenith and viewing angles), enabling comparisons over time. These satellites typically orbit at an altitude between 600 km and 800 km [165].

  In Figure A.11, the Sentinel-2C satellite is shown flying in an SSO at an altitude of 775 km, along with the EnMAP satellite, which operates at an altitude of 653 km.



(a) Sentinel-2C in orbit.
Image credit: European Comission [166].

(b) EnMAP in orbit.
Image credit: OHB and DLR [167].

Figure A.11: Spaceborne vehicles.

The principal advantages of spaceborne sensors are their wide coverage and long operational lifespan, allowing years of continuous monitoring, making them perfectly suitable for global applications. The disadvantages include low spatial resolution (compared to airborne platforms) and the presence of atmospheric elements (e.g. clouds) that may obstruct the view.

- **Airborne Sensors**: Sensors mounted on aircraft at low altitudes offer some advantages compared to spaceborne sensors. For instance, if the vehicle is flying low enough, it can avoid having its view blocked by clouds. In addition, they provide a small ground sampling distance (on the order of centimeters) at the expense of a narrow ground swath.

  These platforms face a new challenge: geometric distortions caused by instabilities in aircraft pitch, roll, and yaw motions during flight, in addition to changes in altitude and velocity during capture [168]. Therefore, a specific geometric rectification is required. Two of the most commonly used vehicles are listed and described below:

  - Airplane: This type of aircraft provides a relatively large coverage range (hundreds of kilometers), meeting the requirements for applications at regional or national levels. It allows continuous maintenance and calibration of the onboard sensor in a controlled environment, at the expense of the associated costs of operating an airplane and the required infrastructure.

    In Figure A.12, an illustration of the platform used for the airborne HySpex [169] hyperspectral sensor of DLR is shown.
  - UAV: Commonly known as drones, they are one of the most cost-effective options for monitoring with high spatial resolution, but they are limited in

(a) DLR Research aircraft - (Dornier) DO 228-212.

(b) HySpex sensor system onboard DO 228-212.

Figure A.12: HySpex hyperspectral sensor onboard DO 228-212 in DLR Oberpfaffenhofen site. Images credit: DLR, CC BY-NC-ND 3.0 [156].

covered distance and sensor weight. A wide range of drone types is available, including multi-rotor, single-rotor, and fixed-wing aircraft. Commercial options are widely available on the market, allowing companies to handle local-level applications with complete image processing software suites. In recent years, these platforms have become very popular for precision agriculture, urban planning, and development. In Figure A.12, an illustration of multi-rotor and fixed-wing UAVs is shown.

- **Ground-based**: Suitable for on-site or laboratory applications, these sensors are designed to remain static or have limited mobility. Given the limited area covered per measurement, they are usually intended for validation campaigns of airborne or spaceborne sensors or for monitoring fixed high-importance sites.

  For example, handheld spectrometers allow for highly accurate measurements in a controlled environment, following designated measurement protocols to minimize uncertainty. For reflectance validation campaigns, measurements must be performed by sampling the site in an equidistant pattern over a uniform surface (e.g., a grass field). The measured site should be larger than the ground sampling distance to mitigate the effect of variance between pixels and to take the average of a set of pixels.

  In Figure A.14, a spectrometer and an example of its application are shown.

Because of the scope of this thesis, not all remote sensing platforms are reviewed. However, sensors can be mounted on a wide range of platforms, such as space shuttles, helicopters, hot air and tethered balloons, ships with sensors above and below the water surface, elevated (raised) platforms, or meteorological stations.

(a) Multi-rotor (quadcopter) drone:
DJI Mavic 3M with a multispectral and RGB camera onboard. Also includes a light sensor on the top for sensor calibration and GPS for geolocalization.
Image credit: DJI.

(b) Fixed-wing drone:
AgEagle eBee X series with modular design for adapting a wide variety of sensors and components depending the application. Image credit: AgEagle.

Figure A.13: Different types of UAVs.

(a) Spectra Vista Corporation (SVC) spectrometer performing the measurement of a reference panel (white square below). The objective is to characterize the incoming solar irradiance, already filtered by the atmosphere. Image credit: Spectravista.

(b) In-situ measurements for PACO-L2A Validation campaign of Sentinel-2 and EnMAP missions. Klosterwiese, Fürstenfeldbruck, Bavaria.

Figure A.14: Hand-held spectrometers application examples.

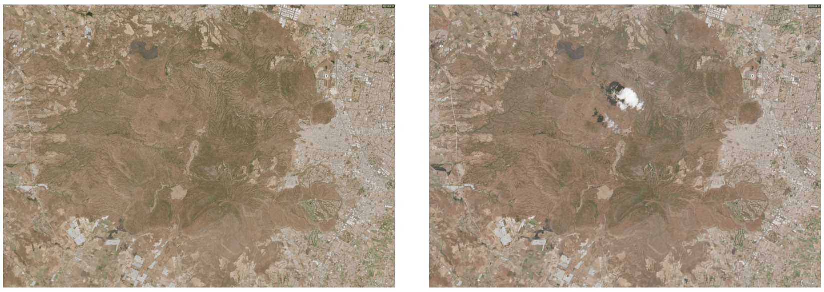
A.1.2.6 Temporal Resolution

Multiple captures of the same location at a constant time interval are key for applications that detect changes (e.g., agriculture, natural disaster damage assessment, defense). In addition, for satellite imagery, where the atmosphere may obstruct ground reflectance retrieval, it is important to capture images within a reasonably short time interval.

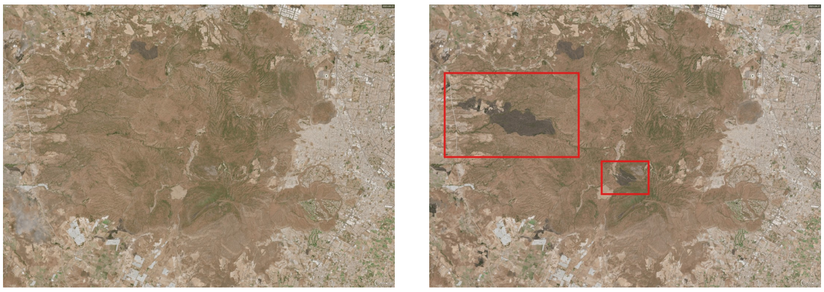
The time between captures is called the *revisit time*. For an orbital sensor, it depends on the altitude, velocity, and FOV of the satellite, typically involving a trade-off between the revisit time and the spatial resolution. Revisit time can also be reduced by adding additional sensors to the constellation with the same characteristics and orbit but phased at a specific angle. For instance, an extra sensor in the same orbit but phased 180° apart would reduce the revisit time by half.

For airborne sensors, their revisit time completely depends on the assigned budget for monitoring a location, generally covering a considerably smaller area with high spatial resolution and not dependent of the cloud coverage.

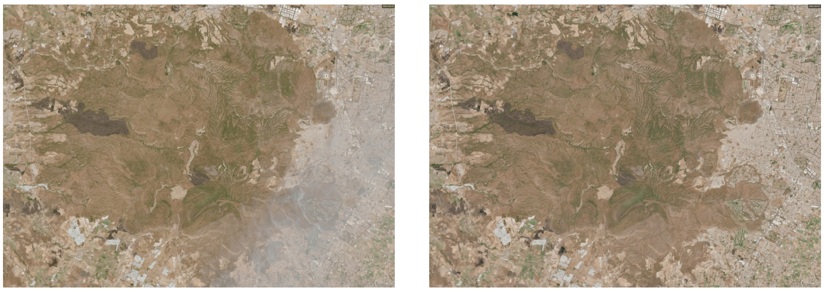
In Figure A.15, an example of a wildfire damage assessment is shown. A high revisit time increases the opportunity to detect potential dangers earlier and improves decision-making by authorities to counter the hazards.



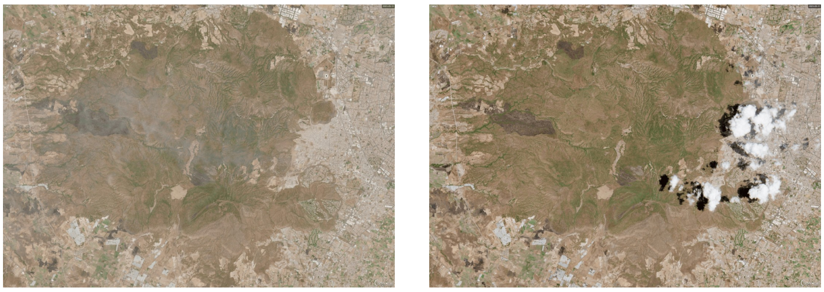
(a) April 12, 2023 (b) April 17, 2023



(c) April 22, 2023 (d) April 27, 2023



(e) May 2, 2023 (f) May 7, 2023



(g) May 12, 2023 (h) May 17, 2023

Figure A.15: One-month temporal series of Sentinel-2 RGB composites with a 5-day revisit time showing Bosque La Primavera, located on the outskirts of Guadalajara, Jalisco, Mexico. A wildfire event started on April 26, 2023, with burn scars clearly visible in Sentinel-2 imagery. Image credit: Copernicus data space.

A.1.2.7 Noise Model

Noise is intrinsic to any signal, and the signal captured by an optical sensor for spectral imaging is not an exception. There are many sources and types of noise present in the captured signal, but in general, noise can be categorized into two classes [170]: fixed pattern noise and random noise.

The first type arises from calibration errors or sensor defects and is not addressed in this work. Instead, random noise, due to its stochastic nature, can be studied and modeled using a suitable noise model.

For new-generation imaging spectrometers used in optical remote sensing, random noise mainly originates from two sources: Signal-Dependent (SD) photonic noise and Signal-Independent (SI) electronic noise, also known as thermal (Johnson) noise [171].

Using tensor theory, the noisy spectral image is represented as a sum of the clean signal and additive noise [172].

$$\mathcal{X}_{noisy} = \mathcal{X} + \mathcal{N}, \tag{A.20}$$

where $\mathcal{X}_{noisy}, \mathcal{X}, \mathcal{N} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. $\mathcal{X}_{noisy}$ is the noisy spectral image, $\mathcal{X}$ is the clean signal and $\mathcal{N}$ is the noise for both photon and thermal noise [170]. The noise model in equation (A.20) is valid under the assumption of high Signal-to-Noise Ratio (SNR) of $\mathcal{X}$. The variance of the noise depends of each pixel value x_{i_1,i_2,i_3} in the clean signal $\mathcal{X}$. The tensor $\mathcal{N}$ is composed by the sum of two tensors, the photonic noise tensor $\mathcal{P} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ and the thermal noise tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Thus:

$$\mathcal{N} = \mathcal{P} + \mathcal{T}, \tag{A.21}$$

where $\mathcal{P}$ is dependent of the clean signal $\mathcal{X}$, and $\mathcal{T}$ is signal-independent.

The improvement of the CCD sensors for new generation instruments exhibited a tendency to increase spatial resolution. Therefore, the number of photons that reach a pixel per unit time becomes smaller, causing the random fluctuation of photons arriving at the sensor. Consequently, the photonic noise is now more relevant than before [170]. Photonic noise follows Poisson distribution [173], but it can be approximated by a Gaussian distribution [174]. A single photon noise element p_{i_1,i_2,i_3} of tensor $\mathcal{P} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ can be expressed in terms of its corresponding element x_{i_1,i_2,i_3} of the clean signal $\mathcal{X}$ as follows [174]:“

$$p_{i_1,i_2,i_3} = (x_{i_1,i_2,i_3})^{\gamma} \cdot u_{i_1,i_2,i_3}, \tag{A.22}$$

where u_{i_1,i_2,i_3} is a stationary, zero-mean uncorrelated random process independent of x_{i_1,i_2,i_3} with variance σ_u^2”. “In the case for earth remote sensing images captured by instruments mounted in airborne or spaceborne platforms, the exponent γ is equal to 0.5” [171]. Thus:

$$p_{i_1,i_2,i_3} = \sqrt{x_{i_1,i_2,i_3}} \cdot u_{i_1,i_2,i_3}. \tag{A.23}$$

The thermal agitation of the charge carriers inside the electronics of the instruments used for spectral images causes the thermal noise. A single thermal noise element of the noise tensor $\mathcal{T}$ is denoted by t_{i_1,i_2,i_3}, this random process can be modeled as an additive zero-mean white Gaussian noise with variance σ_t^2 [170].

From equations (A.20) and (A.21), the noise model considering both photonic and thermal noise is:

$$\mathcal{X}_{noisy} = \mathcal{X} + \mathcal{P} + \mathcal{T}. \tag{A.24}$$

Element-wise, using equations (A.22) and (A.23), the noise model is:

$$x_{i_1,i_2,i_3}^{noisy} = x_{i_1,i_2,i_3} + \sqrt{x_{i_1,i_2,i_3}} \cdot u_{i_1,i_2,i_3} + t_{i_1,i_2,i_3}. \tag{A.25}$$

To highlight dependency, another useful notation for equation (A.21) is [175]:

$$\mathcal{N}(\mathcal{X}) = \mathcal{N}_{SD}(\mathcal{X}) + \mathcal{N}_{SI}. \tag{A.26}$$

$$n_{i_1,i_2,i_3}(\mathcal{X}) = \sqrt{x_{i_1,i_2,i_3}} \cdot u_{i_1,i_2,i_3} + t_{i_1,i_2,i_3}. \tag{A.27}$$

Given this, equation (A.24) can be rewritten as:

$$\mathcal{X}_{noisy}(\mathcal{X}) = \mathcal{X} + \mathcal{N}_{SD}(\mathcal{X}) + \mathcal{N}_{SI}. \tag{A.28}$$

To get the noise variances of the random processes σ_{u,i_3}^2 and σ_{t,i_3}^2, are required the mean variance of the noise tensors $\mathcal{N}_{SD}(\mathcal{X})$ and $\mathcal{N}_{SI}$ [175]. For SD mean noise variance tensor:

$$\sigma_{\mathcal{N}_{SD}(\mathcal{X})}^2 = \frac{1}{I_1 I_2 I_3} \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \sigma_{u,i_3}^2 \cdot x_{i_1,i_2,i_3}. \tag{A.29}$$

Let μ_{i_3} be the mean of the clean signal at band i_3:

$$\mu_{i_3} = \frac{1}{I_1 I_2} \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} x_{i_1,i_2,i_3}, \tag{A.30}$$

from (A.30), equation (A.29) can be rewritten as:

$$\sigma_{\mathcal{N}_{SD}(\mathcal{X})}^2 = \frac{1}{I_3} \sum_{i_3=1}^{I_3} \sigma_{u,i_3}^2 \cdot \mu_{i_3}, \tag{A.31}$$

besides, the SI noise has constant variance $\sigma_{\mathcal{N}_{SI}}^2$ in all bands, the SI mean variance noise tensor is:

$$\sigma_{\mathcal{N}_{SI}}^2 = \frac{1}{I_1 I_2 I_3} \sum_{i_3=1}^{I_3} \sigma_{t,i_3}^2, \tag{A.32}$$

thus, the mean variance of the noise tensor $\mathcal{N}(\mathcal{X})$ is:

$$\sigma_{\mathcal{N}(\mathcal{X})}^2 = \sigma_{\mathcal{N}_{SD}(\mathcal{X})}^2 + \sigma_{\mathcal{N}_{SI}}^2, \tag{A.33}$$

$$\sigma^2_{\mathcal{N}(\mathcal{X})} = \frac{1}{I_1 I_2 I_3} \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \left(\sigma^2_{u,i_3} \cdot x_{i_1,i_2,i_3} + \sigma^2_{t,i_3} \right). \tag{A.34}$$

From the SNR (dB) formula:

$$\mathrm{SNR} = 10 \cdot \log_{10} \frac{\|\mathcal{X}\|^2}{\|\mathcal{N}\left(\mathcal{X}\right)\|^2}, \tag{A.35}$$

$\|\mathcal{N}\left(\mathcal{X}\right)\|^2$ in terms of $\mathcal{X}$ and a specified SNR is expressed as:

$$\|\mathcal{N}\left(\mathcal{X}\right)\|^2 = \|\mathcal{X}\|^2 \cdot 10^{-\left(\frac{\mathrm{SNR}}{10}\right)}. \tag{A.36}$$

If equation (A.36) is divided by the total number of pixels $I_1 I_2 I_3$, note that $\sigma^2_{\mathcal{N}(\mathcal{X})} = \frac{\|\mathcal{N}(\mathcal{X})\|^2}{I_1 I_2 I_3}$ (see equation (A.34)). If $\overline{P}_{\mathcal{X}} = \frac{\|\mathcal{X}\|^2}{I_1 I_2 I_3}$ is the mean power of tensor $\mathcal{X}$, then:

$$\sigma^2_{\mathcal{N}(\mathcal{X})} = \overline{P}_{\mathcal{X}} \cdot 10^{-\left(\frac{\mathrm{SNR}}{10}\right)}. \tag{A.37}$$

Assuming a parameter α, which control the dominance of signal-dependent noise variance over the signal-independent noise variance, such that:

$$\alpha = \frac{\sigma^2_{\mathcal{N}_{SD}(\mathcal{X})}}{\sigma^2_{\mathcal{N}_{SI}}}, \tag{A.38}$$

Then, from equations (A.38) and (A.33), follows:

$$\sigma^2_{\mathcal{N}_{SD}(\mathcal{X})} = \frac{\sigma^2_{\mathcal{N}(\mathcal{X})} \cdot \alpha}{\alpha + 1}, \tag{A.39}$$

and:

$$\sigma^2_{\mathcal{N}_{SI}} = \frac{\sigma^2_{\mathcal{N}(\mathcal{X})}}{\alpha + 1}. \tag{A.40}$$

Note that both results depend only on α and $\sigma^2_{\mathcal{N}(\mathcal{X})}$, which are already available in equations (A.37, A.38). Finally, solving for the noise variance of the random process σ^2_{u,i_3} from equation (A.29):

$$\sigma^2_{u,i_3} = \frac{\sigma^2_{\mathcal{N}_{SD}(\mathcal{X})}}{\mu_{i_3}}, \tag{A.41}$$

as well, the noise variance of the random process σ^2_{t,i_3}, from equation (A.31):

$$\sigma^2_{t,i_3} = \sigma^2_{\mathcal{N}_{SI}}. \tag{A.42}$$

A.2 Remote Sensing Missions

In the 2022 study by Qiang Zhao et al. [157], an overview of the exponentially growing number of publications is provided, highlighting the most cited EO missions, as shown in Figure A.16.

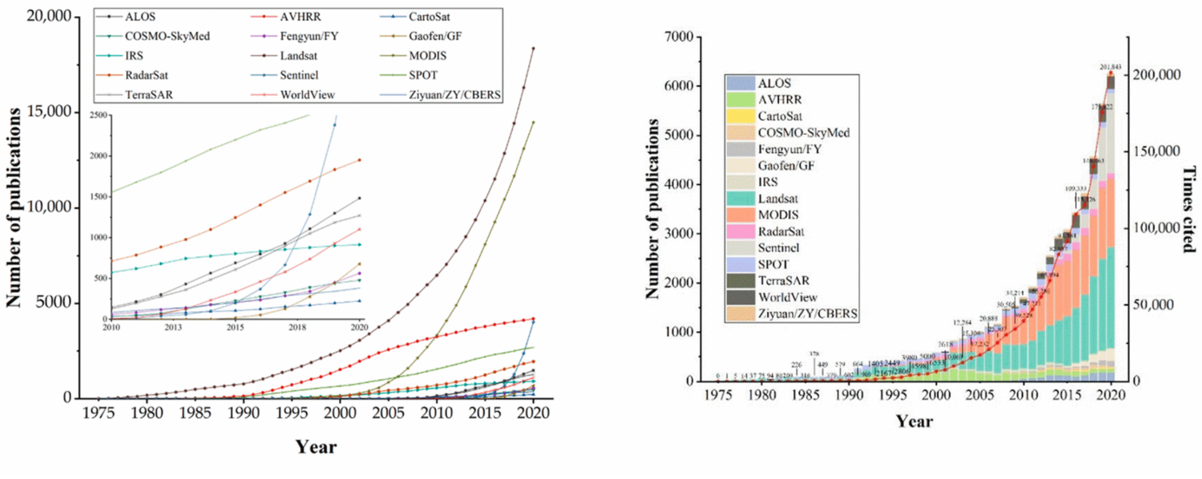


Figure A.16: Number of publications referencing the use of data from an EO mission. Left: Number of publications per mission per year. Right: Total number of publications per year. Plot credit: Qiang Zhao et al. [157]

Additionally, the study provides the distribution of publications by country. The leading countries are the USA (14,003) and China (11,165), followed by Germany (2,960), Canada (2,760), India (2,743), and other European countries. Mexico is still developing in this area, with 488 publications, highlighting opportunities for advancing solutions at the national level. The worldwide distribution is shown in Figure A.17.

The derived products from Earth monitoring missions cover a wide range of applications of interest to both the private and public sectors. For instance, the EU has developed the Copernicus Program, a collection of Earth monitoring missions with six principal application areas, as listed in Figure A.18.

Climate change monitoring is becoming one of the largest research areas and has initiated a collaborative effort among public institutions worldwide, given humanity's shared interest in mitigating the effects of global warming.

Resource management for land and water is of interest to governments and primary sector companies, with objectives such as food security and water management. It is also essential for monitoring natural resources, ensuring their sustainable exploitation, and promoting responsible waste disposal.

Atmospheric monitoring allows for tracing harmful gases emitted by industries and cities, as well as forecasting weather, which is closely related to managing emergencies

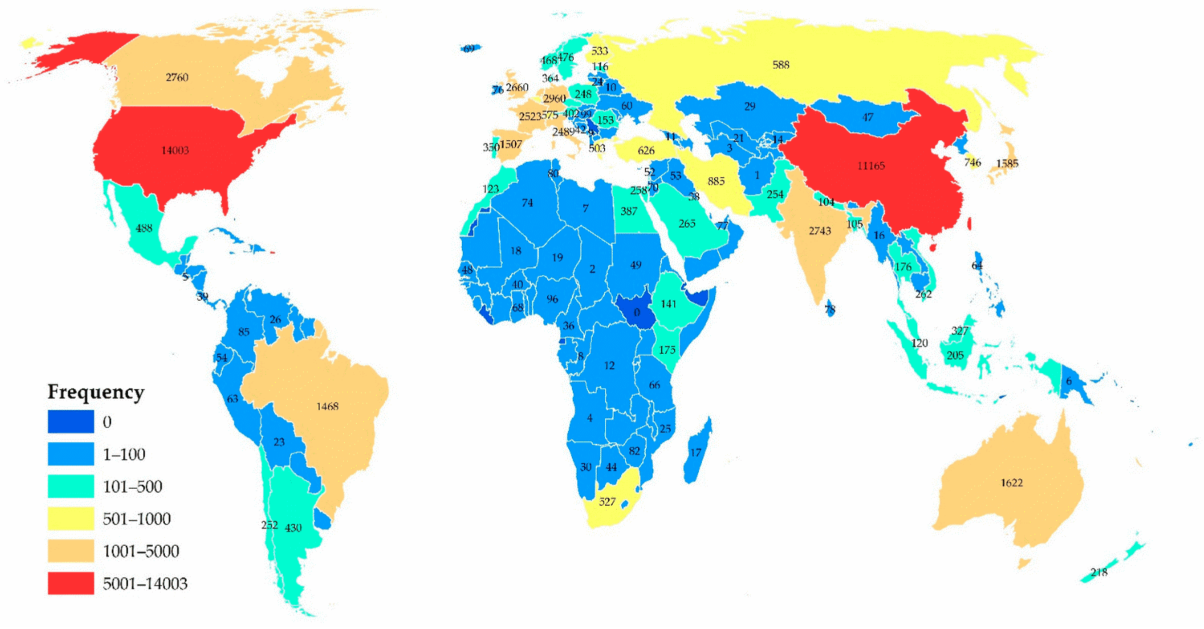


Figure A.17: Worldwide distribution of publications referencing the use of data from EO missions. Image credit: Qiang Zhao et al. [157].

Figure A.18: The six missions from EU's Copernicus Program. Image credit: Copernicus Program, EU.

and natural disasters, such as hurricanes, floods, and earthquakes. These capabilities aid protection authorities in preparing for events and assessing damages afterward. The retrieved products from EO missions enable informed decision-making and planning.

Lastly, security and defense is a growing sector due to the world's geopolitical context and the increasing budgets allocated by nations to this area, representing a significant advantage for defense institutions given the vast amount of retrieved information.

All the mentioned areas benefit from the geoinformation retrieved by EO missions. In this thesis, data captured from a multispectral sensor (Sentinel-2) and a hyperspectral sensor (EnMAP) is used for training the models in the experiments presented. On the other hand, cloud products from the TROPOMI sensor onboard the Sentinel-5P mission are used for the testing phase of the resulting models.

The selection of these sensors is due to the fact that they are processed by PACO for atmospheric correction, and some products derived from the early stages of processing are used to create training datasets for this work. In addition, the data are publicly available for research purposes.

A.3 Spectral Indices

Spectral indices highlight physical characteristics of the pixel composition, and there exists an extensive variety of spectral indices for a wide range of applications, including land cover classification, vegetation identification, cryosphere monitoring, cloud detection, resource mapping, among many others. To illustrate how spectral indices work, the following three examples are selected from a wide variety of indices for deeper analysis.

The Normalized Difference Vegetation Index (NDVI) [176] measures vegetation health, given that vegetation reflects strongly in the NIR $\sim 850\,\mathrm{nm}$ and absorbs in the Red $\sim 660\,\mathrm{nm}$. Negative values correspond to water, and values around 0 to rock, sand, or snow. NDVI is defined as follows:

$$\mathrm{NDVI} = \frac{\mathrm{NIR} - \mathrm{Red}}{\mathrm{NIR} + \mathrm{Red}} \in [-1, 1]. \tag{A.43}$$

The Normalized Difference Water Index (NDWI) [177] highlights water features and suppresses soil and vegetation, given that water reflects in the Green $\sim 560\,\mathrm{nm}$ and absorbs in the NIR $\sim 850\,\mathrm{nm}$. NDWI is defined as follows:

$$\mathrm{NDWI} = \frac{\mathrm{Green} - \mathrm{NIR}}{\mathrm{Green} + \mathrm{NIR}} \in [-1, 1]. \tag{A.44}$$

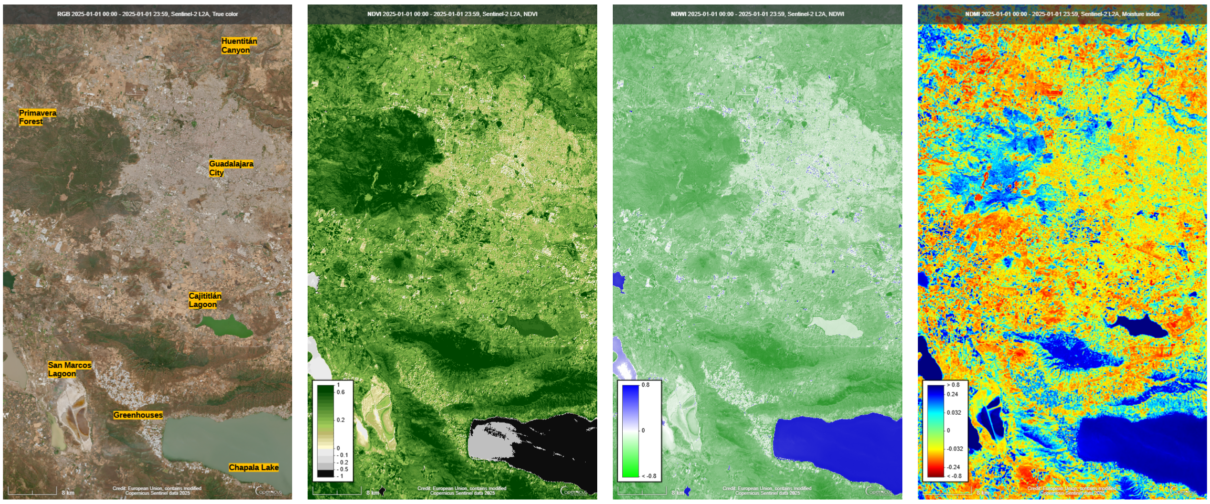
The Normalized Difference Moisture Index (NDMI) [178] detects water content in vegetation and overall surface moisture, given that water in vegetation absorbs more in the SWIR $\sim 1640\,\mathrm{nm}$ than in the NIR $\sim 850\,\mathrm{nm}$. Values around 0 correspond to vegetation with water stress, and negative values to non-vegetated soil. NDMI is defined as follows:

$$\mathrm{NDMI} = \frac{\mathrm{NIR} - \mathrm{SWIR}}{\mathrm{NIR} + \mathrm{SWIR}} \in [-1, 1]. \tag{A.45}$$

The three selected spectral indices, NDVI, NDWI, and NDMI, are displayed in Figure A.19, along with their RGB true color composite for comparison, highlighting Guadalajara city (top right), Primavera Forest (top left), and Chapala Lake (bottom right), with the Cajititlán Lagoon to its north (the smaller, greener lagoon).

Some insights related to vegetation can be analyzed using the NDVI map, where a green spot can be identified in the Primavera Forest, along with other patches far from the city and closer to the lake. Although the NDVI indicates healthy vegetation, the NDMI reveals some water stress spots in the middle of the forest. Another observation is that within the city, vegetation is more present in the northwest area. One possible application of these insights could be to support the planning of an urban reforestation campaign within Guadalajara city.

In addition, insights about water bodies and water quality can be analyzed using the NDWI and NDMI. Although Chapala Lake is well highlighted in the NDWI map,



(a) RGB. (b) NDVI. (c) NDWI. (d) NDMI.

Figure A.19: True color and example spectral index maps. Image: Sentinel-2, over Guadalajara and surrounding areas, captured on 2025/01/01. Source: Copernicus Data Workspace.

the Cajititlán Lagoon is missed, but it is clearly visible in the NDMI and also in the NDVI, which may be attributed to dense aquatic vegetation or shallow waters. This information could be used to monitor water levels over time using a time series, supporting water management and monitoring applications. Furthermore, although the San Marcos Lagoon appears dry in the RGB image and shows low NDWI values, the moisture index provided by the NDMI reveals that the water content of the area is high. Additionally, the highways show consistently low moisture, which means that in the case of a flood, these areas would be expected to show higher values.

With this brief analysis, an intuitive understanding of the strengths and weaknesses of spectral indices is provided, but they should be further interpreted with expert knowledge. Although only three spectral index examples are analyzed in this thesis, multiple application areas use different spectral indices that denote particular characteristics present in the spectral signature. Some examples are listed in Table A.1.

Index	**Formula**	**Application Area**
NDVI [176]	$\frac{NIR-Red}{NIR+Red}$	Vegetation health and biomass
NDWI [177]	$\frac{Green-NIR}{Green+NIR}$	Water body detection
NDMI [178]	$\frac{NIR-SWIR}{NIR+SWIR}$	Vegetation moisture content
NDBI [179]	$\frac{SWIR-NIR}{SWIR+NIR}$	Urban and built-up area detection
SAVI [180]	$\frac{(NIR-Red)(1+0.5)}{NIR+Red+0.5}$	Vegetation with soil adjustment
EVI [181]	$\frac{2.5(NIR-Red)}{NIR+6\cdot Red-7.5\cdot Blue+1}$	Dense vegetation / canopy structure
MSAVI [182]	$\frac{2\cdot NIR+1-\sqrt{(2\cdot NIR+1)^2-8\cdot(NIR-Red)}}{2}$	Vegetation, soil minimized
BAI [183]	$\frac{1}{(Red-0.06)^2+(NIR-0.1)^2}$	Burned area detection
NDSI [184]	$\frac{Green-SWIR}{Green+SWIR}$	Snow cover and cloud-snow separation
Cloud ratio	$\frac{Blue}{SWIR}$	Cloud detection

Table A.1: Spectral indices and band ratios for optical remote sensing applications.

It is important to note that the quantities retrieved from spectral indices should be further interpreted and analyzed, considering the possibility of “false positives.” An example is shown in Figure A.20, where white greenhouses show high values (in grayscale) using the cloud ratio (described in Table A.1). During cloud masking, these greenhouses could easily be classified as clouds if another spectral index or external information is not included in the masking process.

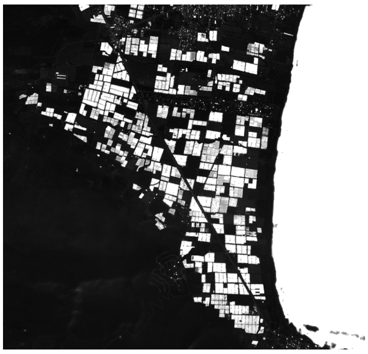


Figure A.20: Cloud ratio.

A.4 Earth's Spheres and Its Classification

This section describes all the classes that are of common interest to the remote sensing research community. In the context of developing the ground segment processing chain of an remote sensing mission, the requirements of a wide range of research areas must be considered during processing. E.g., land cover identification, water quality analysis, atmospheric and trace gas analysis, vegetation monitoring, cryosphere studies, among many others.

Pixel classification performed at the L1C or L2 level allows for quick identification of pixels of interest in subsequent processing steps. For example, during water analysis, all land-related pixels are not relevant and should be discarded.

The following description does not aim to define Earth's spheres within their respective research domains. On the contrary, its purpose is to highlight the most important targets for spectral imagery classification, addressing only the characteristics relevant to this thesis in the context of optical remote sensing and AC. The main applications related to the observation of these targets are discussed in Section 2.2.

A.4.1 Lithosphere, Hydrosphere and Cryosphere

The *lithosphere* comprises the outer solid layer of Earth's surface. For the purpose of this thesis, it also includes all visible land surfaces, encompassing the biosphere and human-built structures. This class is referred to as *clear sky land*, or simply *land*, and is further subdivided into *vegetated*, *non-vegetated*, or *built-up* areas. An example of the aforementioned classes is shown in Figure A.21, capturing the municipality of "Tepatitlán de Morelos" (built-up), surrounded by agricultural crops (some vegetated, others already harvested, leaving the bare soil exposed).

Credit: European Union, contains modified Copernicus Sentinel data 2025

Figure A.21: "Altos de Jalisco", Mexico. Image: Sentinel-2 captured on 2024/10/03.

The *hydrosphere* includes all the water present on Earth. For remote sensing, the visible water on Earth's surface comprises oceans, rivers, lakes, and lagoons, including human-built water reservoirs. The *water* class can be further subdivided based on salinity, depth, vegetation, or pollution. In Figure A.22, Chapala Lake (the largest lake in Mexico and the main source of water supply for the city of Guadalajara) is shown with visible algal bloom proliferation due to pollution and low water levels. This image was selected as the Copernicus "Image of the Day" on February 2024 [185].

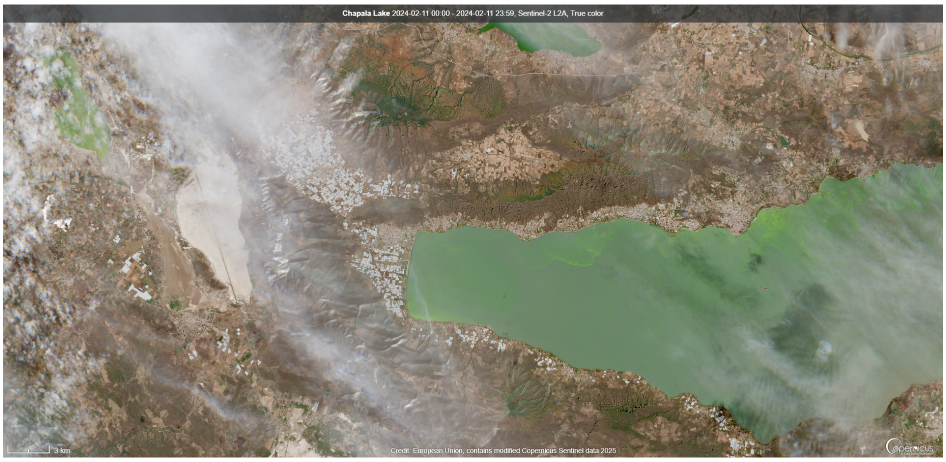


Figure A.22: Chapala Lake, Jalisco, Mexico. Image: Sentinel-2 captured on 2024/02/11.

The cryosphere comprises all temporal and permanent *snow* and *ice*, including *glaciers*. These are mostly found near the Earth's poles (on land or floating on water) and in high-altitude mountains, and are highly dependent on the season. In Figure A.23, the Eastern Alps are shown, with Innsbruck located in the middle of the Inn Valley.

Figure A.23: Innsbruck, Austria. Image: Sentinel-2 captured on 2024/12/28.

A.4.2 Atmosphere

The *atmosphere* is composed of aerosols and gases distributed at different altitudes. The *troposphere* (from Earth's surface to 15 km) contains the densest part of the atmosphere, including all aerosols and the majority of gases. The *stratosphere* (15 km–50 km) contains significant quantities of ozone. The *mesosphere* (50 km–85 km) and the *thermosphere* (85 km–600 km), where satellites orbit, contain much lower gas densities [186]. The principal components that absorb and scatter sunlight are found in the *troposphere* (WV, CO_2, Methane (CH_2), O_3, Nitrous Oxide (N_2O), and O_2) and the *stratosphere* (primarily O_3).

Clouds are the primary obstacle in optical remote sensing, as they reflect the majority of sunlight across the EM spectrum (which gives them their white appearance by reflecting the VIS range with approximately equal intensity). They behave differently depending on their altitude (CTH), exhibiting variations in shape, spatial extent (CF), color, and transparency, all of which directly affect how they interact with sunlight (COT). In Figure A.24 a diagram is showing describing shape and color of common types of clouds at different altitudes. In addition, in Table A.2, a brief description of each kind of cloud is listed as follows:

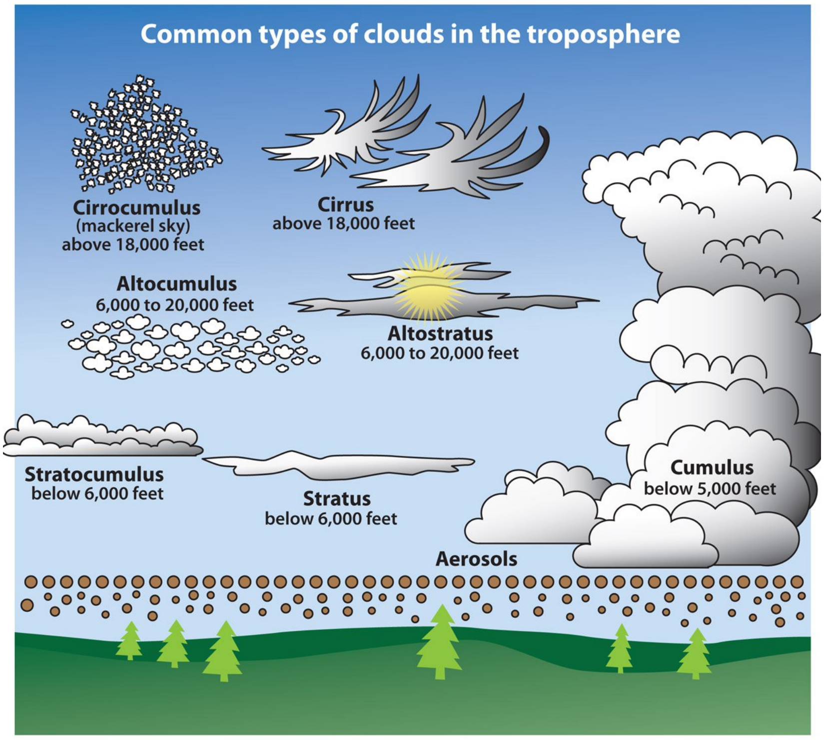


Figure A.24: Common types of clouds. Image credit: University Corporation for Atmospheric Research [158].

Cloud Type	Composition	Key Bands	Interactions with EM Spectrum
Cirrus (Above 5,500 m)	Ice crystals	1.38 μm, 8.5–12 μm	High reflectance in 1.38 μm due to cirrus altitude above water vapor. Often semi-transparent in VIS. [187, 188]
Cirrocumulus (Above 5,500 m)	Ice, supercooled water	VIS, NIR	Thin cloud layer; weak reflectance signal. Causes minor scattering at high altitudes, with limited TOA impact. [189]
Altocumulus (2,000–6,000 m)	Liquid water, mixed-phase	0.86 μm, 1.6 μm	Strong scattering in VIS; moderate reflectance in SWIR due to droplet size. [188, 190]
Altostratus (2,000–6,000 m)	Water + ice mix	6.7 μm, 11 μm	Thick, multi-layered; absorbs across thermal IR. Significantly reduces surface visibility. [189]
Stratocumulus (Below 2,000 m)	Liquid water	VIS, NIR	High reflectance in VIS/NIR; strongly affects surface reflectance estimates. [190]
Stratus (Below 2,000 m)	Water droplets	VIS, Thermal IR	Low, thick cloud; strong thermal IR emission. Spectrally similar to fog. [189]
Cumulus (Below 1,500 m)	Liquid water	VIS, NIR, SWIR	Bright reflectance; casts shadows. Commonly masks land in classification algorithms. [188]
Aerosols (Near surface)	Dust, smoke, salt, organics	UV, VIS, NIR	Variable scattering and absorption properties; critical for accurate atmospheric correction. [191, 192]

Table A.2: Characteristics of cloud types and aerosols in optical remote sensing.

In Figure A.25, an example of different types of clouds is shown over the well-known remote sensing site "Fünf Seen" in Bavaria, Germany. Semi-transparent clouds can be observed at high altitudes, allowing thicker clouds at lower altitudes to also be seen.

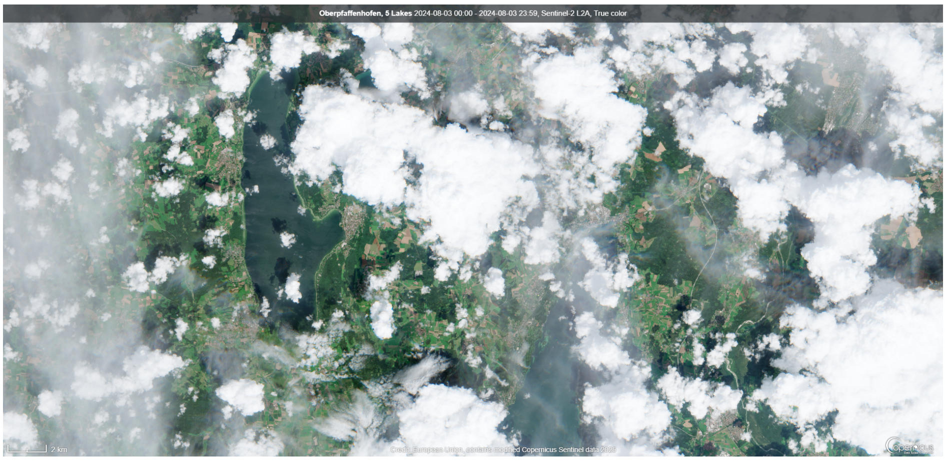


Figure A.25: Starnberg, Bavaria, Germany. Image: Sentinel-2 captured on 2024/08/03.

A.4.3 Shadows

Shadows are dark areas where sunlight is completely absorbed or reflected. This is the case of solid objects (e.g. a mountain) or those with low transmittance (e.g. clouds). Although the surface with the casted shadow is slightly illuminated by reflections from neighboring surfaces or objects, this causes very low reflectance measured by the sensor, which is not sufficient to retrieve spectral information.

In this thesis, shadows are divided into two types depending on the object that blocks or reduces the transmitted light:

- *Cloud shadows*: caused by all types of clouds that cast a shadow over land, water, or snow.
- *Topographic shadows*: caused by mountains, hills, vegetation, or human-built structures.

The size and location of shadows captured by the sensors depend on the solar zenith angle (which determines where the shadows are projected) and the viewing zenith angle (which determines which portion of the shadows is observed). When the sensor zenith angle is 0° (at-nadir capture) and the solar zenith angle is 0°, the shadows are cast vertically and, therefore, are not observed by the sensor.

In Figure A.26, an example is shown of thick clouds casting shadows over land and water in the surroundings of Athens, Greece. Since this scene was captured in the Northern Hemisphere during the winter season, the solar zenith angle is larger, resulting in shadows cast farther from the cloud locations.

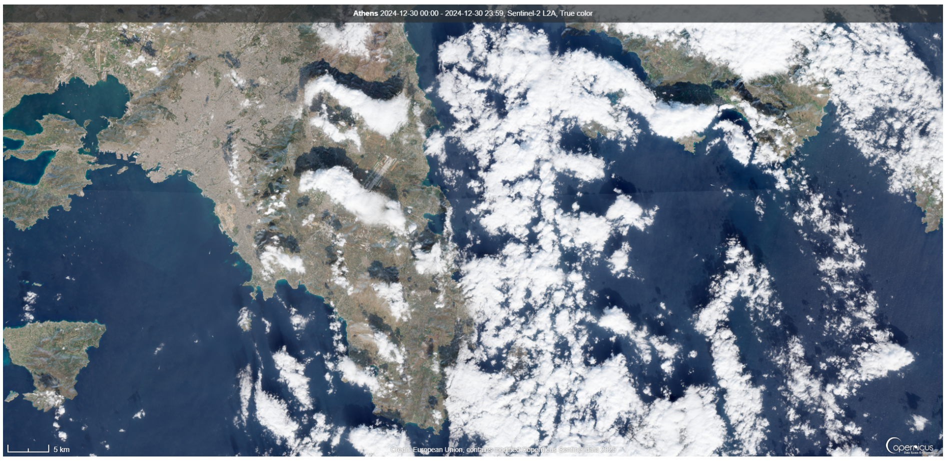


Figure A.26: Athens, Greece. Image: Sentinel-2 captured on 2024/12/30.

This page is intentionally left blank.

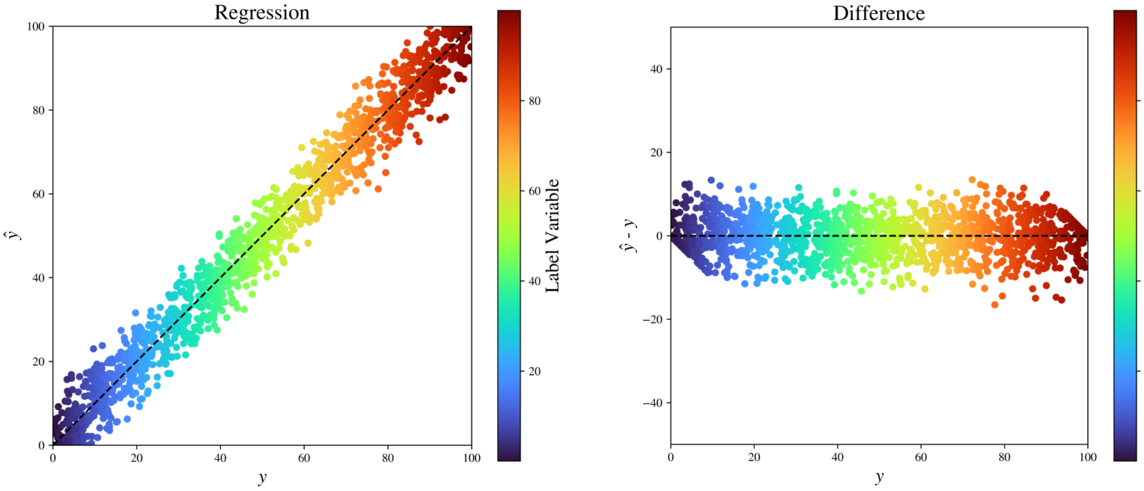
Appendix B

Methodology

B.1 Regression Plots and Histograms

The classic approach to reveal the linear relationship between the reference and evaluated variables is to display both in a regression plot, composed of a scatter plot with y and $\hat{y}$ on the horizontal and vertical axes, respectively. In Figure B.1, a regression plot and its corresponding difference plot between the evaluated variables are shown. The colorbar is used to characterize the type of sample by incorporating a third "label" variable. The plot examples in this subsection are generated using $N = 1500$ synthetic samples from a uniform distribution with additive normal noise, such that $\{y \sim U(a = 0, b = 100)\}$ and $\{\hat{y} = y + \mathcal{N}(\mu = 0, \sigma = 5) \,|\, \hat{y} = \min(\max(\hat{y}, 0), 100)\}$.

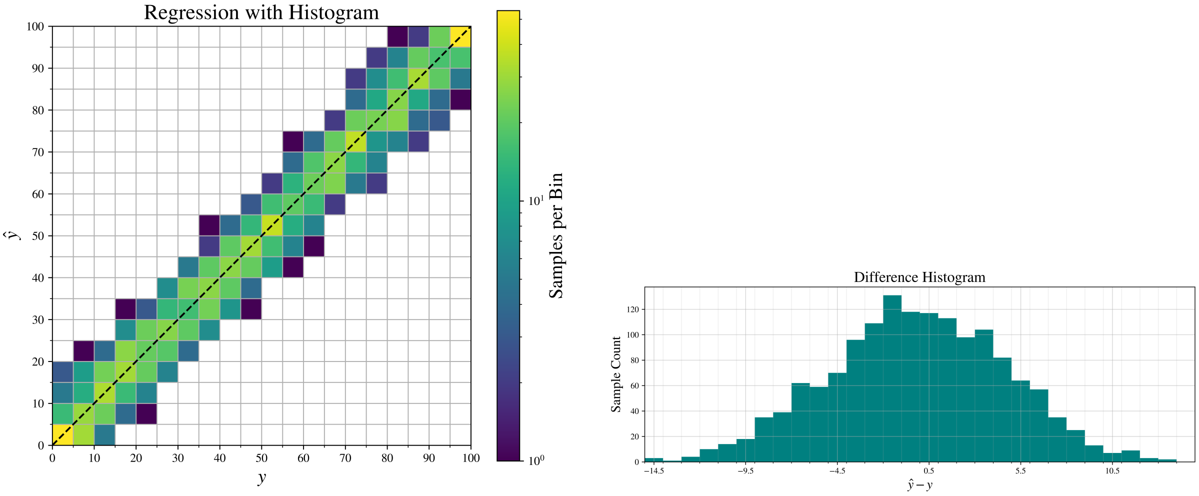
In addition to the regression plot, the use of histograms provides further information about the distribution of samples across different ranges of the variables. For example, the 2D histogram shown in Figure B.2a highlights the sample distribution over the range of the reference and predicted variables. Moreover, the histogram of the differences reveals biases in the predictions, as displayed in Figure B.2b.



(a) Regression plot revealing the linear relationship between y and $\hat{y}$.

(b) Differences between $\hat{y}$ and y.

Figure B.1: Regression and Δ_y plots.



(a) 2D histogram showing the distribution of samples, with color map indicating sample density.

(b) Histogram of differences between y and $\hat{y}$.

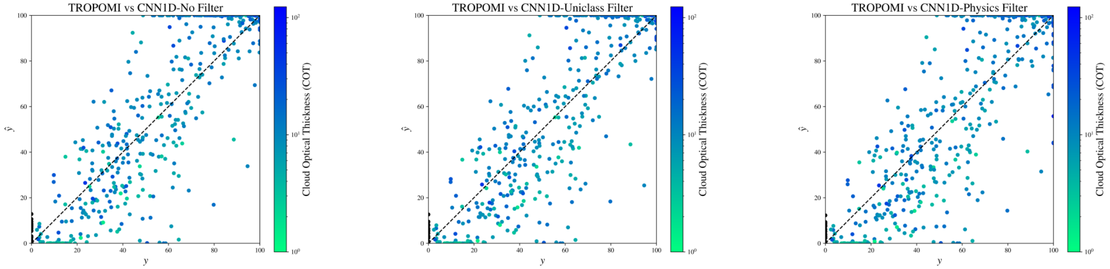
Figure B.2: Histograms.

Appendix C

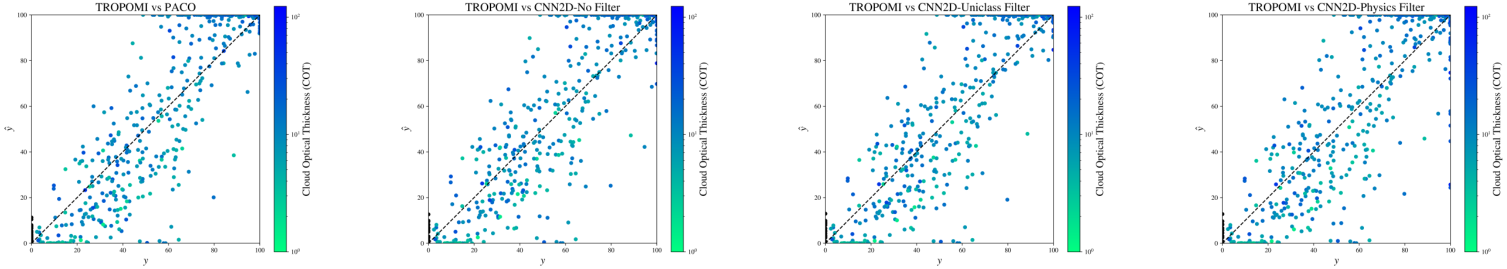
Additional Results

C.1 Results for the Hyperspectral Case

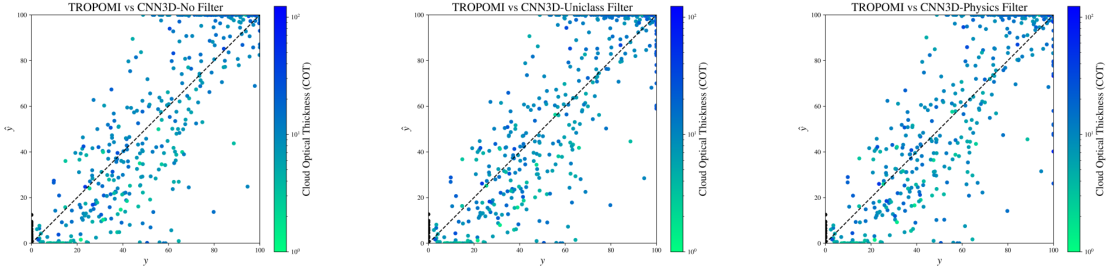
As mentioned in Subsection 4.2.4, this appendix compiles the full set of regression plots and their corresponding difference maps, along with histograms of the differences to better assess the distribution of prediction errors. The regression and difference plots (Figures C.1 and C.2) provide a visual comparison between the CNN-based model predictions and the cloud fraction values derived from TROPOMI. However, due to the clustered nature of the data and the absence of a continuous regression surface, individual data points may be difficult to interpret in detail. The color mapping based on Cloud Optical Thickness (COT) is included to highlight internal patterns in the data, although this information is not directly leveraged in the current analysis. The most informative visualizations in this appendix are arguably the histograms of the prediction differences (Figure C.3), which allow for a clearer evaluation of the distribution and magnitude of model deviations from the reference values.



(a) CNN1D No Filter (b) CNN1D Uniclass (c) CNN1D Physics

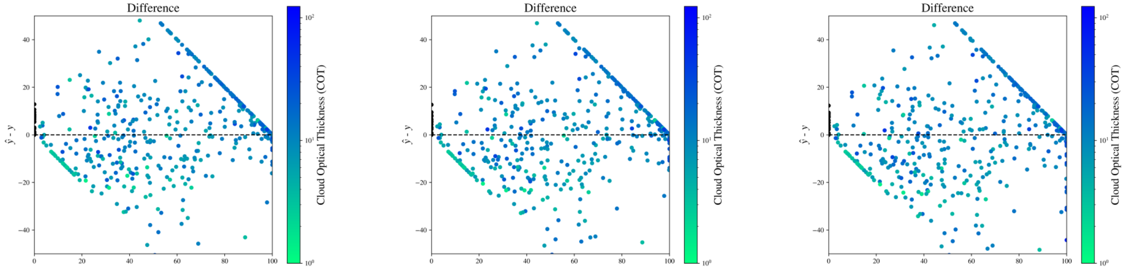


(d) PACO (e) CNN2D No Filter (f) CNN2D Uniclass (g) CNN2D Physics

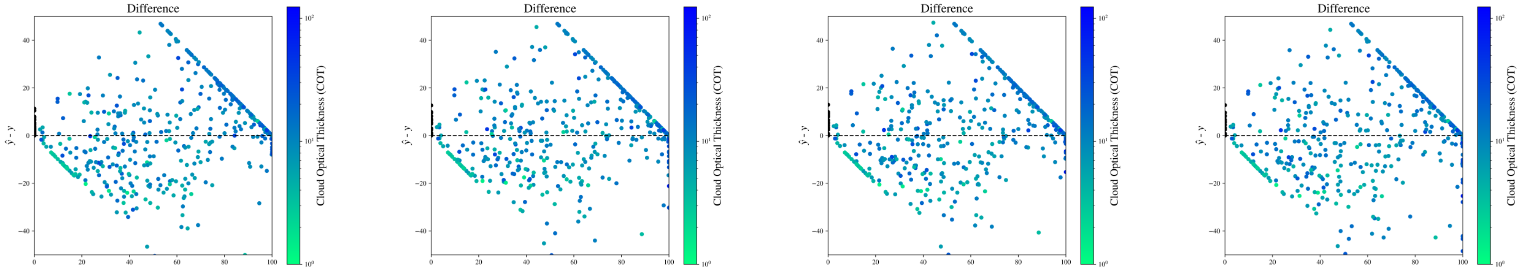


(h) CNN3D No Filter (i) CNN3D Uniclass (j) CNN3D Physics

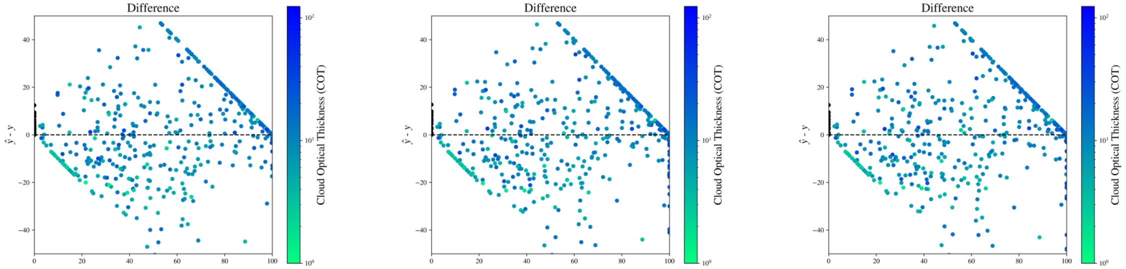
Figure C.1: Regression plots comparing TROPOMI with predictions from all CNN models and filter types.



(a) CNN1D No Filter (b) CNN1D Uniclass (c) CNN1D Physics

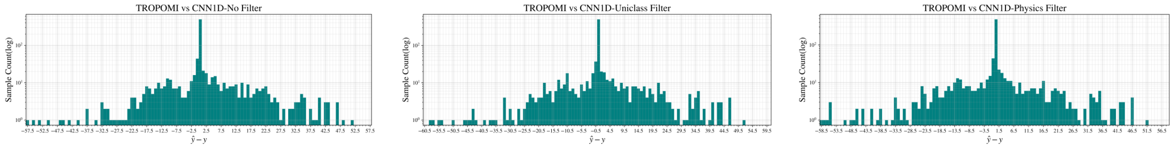


(d) PACO (e) CNN2D No Filter (f) CNN2D Uniclass (g) CNN2D Physics

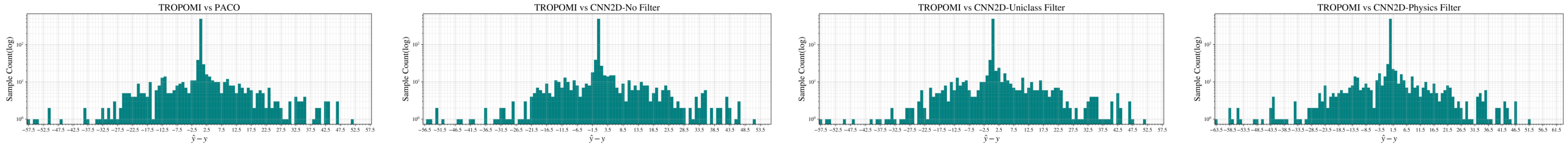


(h) CNN3D No Filter (i) CNN3D Uniclass (j) CNN3D Physics

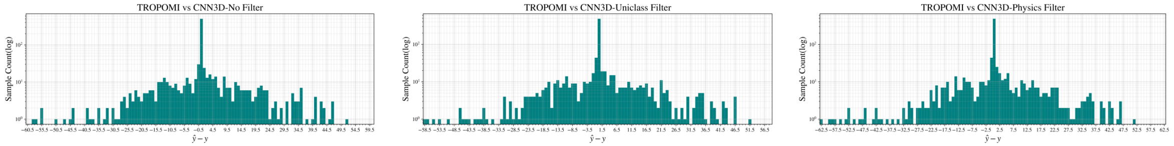
Figure C.2: Difference plots between TROPOMI and CNN predictions for all model architectures and filter types.



(a) CNN1D No Filter (b) CNN1D Uniclass (c) CNN1D Physics



(d) PACO (e) CNN2D No Filter (f) CNN2D Uniclass (g) CNN2D Physics



(h) CNN3D No Filter (i) CNN3D Uniclass (j) CNN3D Physics

Figure C.3: Histogram of differences between TROPOMI and CNN predictions across all models and filter strategies.