Journal Pre-proofs

Review

A critical review on the state-of-the-art and future prospects of Machine Learning for Earth Observation Operations

Pablo Miralles, Kathiravan Thangavel, Antonio Fulvio Scannapieco, Nitya Jagadam, Prerna Baranwal, Bhavin Faldu, Ruchita Abhang, Sahil Bhatia, Sebastien Bonnart, Ishita Bhatnagar, Beenish Batul, Pallavi Prasad, Héctor Ortega-González, Harrish Joseph, Harshal More, Sondes Morchedi, Aman Kumar Panda, Marco Zaccaria Di Fraia, Daniel Wischert, Daria Stepanova

PII: S0273-1177(23)00145-X
DOI: https://doi.org/10.1016/j.asr.2023.02.025
Reference: JASR 16573

To appear in: *Advances in Space Research*

Received Date: 5 August 2022
Revised Date: 10 February 2023
Accepted Date: 13 February 2023

Please cite this article as: Miralles, P., Thangavel, K., Fulvio Scannapieco, A., Jagadam, N., Baranwal, P., Faldu, B., Abhang, R., Bhatia, S., Bonnart, S., Bhatnagar, I., Batul, B., Prasad, P., Ortega-González, H., Joseph, H., More, H., Morchedi, S., Kumar Panda, A., Zaccaria Di Fraia, M., Wischert, D., Stepanova, D., A critical review on the state-of-the-art and future prospects of Machine Learning for Earth Observation Operations, *Advances in Space Research* (2023), doi: https://doi.org/10.1016/j.asr.2023.02.025

COSPAR

A critical review on the state-of-the-art and future prospects of Machine Learning for Earth Observation Operations

Pablo Miralles*, Kathiravan Thangavel*, Antonio Fulvio Scannapieco*, Nitya Jagadam*, Prerna Baranwal*, Bhavin Faldu*, Ruchita Abhang*, Sahil Bhatia*, Sebastien Bonnart*, Ishita Bhatnagar*, Beenish Batul*, Pallavi Prasad*, Héctor Ortega-González*, Harrish Joseph*, Harshal More*, Sondes Morchedi*, Aman Kumar Panda*, Marco Zaccaria Di Fraia*, Daniel Wischert*, Daria Stepanova*.

(*) In respective order:
GTD International, GTD Group, 99 Route d'Espagne, 31100, Toulouse, France, pmirallesr@gmail.com,
Sir Lawrence Wackett Defence & Aerospace Centre, RMIT University, Melbourne, VIC 3083, Australia. kathiravan.thanagvel@student.rmit.edu.au
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, antonio.scannapieco@spacegeneration.org
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, jagadamnitya@gmail.com
Department of Electrical & Electronics/Instrumentation & Department of Mathematics, Birla Institute of Technology and Science (BITS), Pilani, Rajasthan - 333031, India, f2016568@pilani.bits-pilani.ac.in
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, bhavinfaldu6474@gmail.com
Department of Computer Science Engineering, Savitribai Phule Pune University, Pune, Maharashtra - 411052, India, rabhang09@gmail.com
Department of Aerospace Engineering, University of Petroleum and Energy Studies, Energy Acres, Bidholi via Premnagar, Dehradun, Uttarakhand - 248007, India, sahil2112.b@gmail.com
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, sgac@sbonnart.fr
Department of Chemical Engineering & Department of Physics, Birla Institute of Technology and Science (BITS), Pilani, Rajasthan - 333031, India, f20180306@pilani.bits-pilani.ac.in
Institute of Aviation Studies, University of Management and Technology, Lahore, Pakistan, beenish.batul@umt.edu.pk
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, 23. prasad@gmail.com
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, ortegagonzalez.hector@gmail.com
Sapienza University of Rome, Via Eudossiana 18, 00184, Rome, Italy, harrishjagan@gmail.com
Sapienza University of Rome, Via Eudossiana 18, 00184, Rome, Italy, harshal.more@spacegeneration.org
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, sondesmorchedi@gmail.com
ISAE-École Nationale Supérieure de Mécanique et d'Aérotechnique, Téléport 2 – 1 avenue Clément Ader BP 40109 86961 Futuroscope Chasseneuil Cedex, Poitiers, France. aman.panda143@gmail.com
Centre for Electronic Warfare Information and Cyber, Cranfield University, The Defence Academy of The UK, Shrivenham, SN6 8LA, UK, marco.di-fraia@cranfield.ac.uk
Space Generation Advisory Council (SGAC), c/o European Space Policy Institute, Schwarzenbergplatz 6, 1030 Vienna, Austria, daniel.wischert@spacegeneration.org
German Orbital Systems GmbH, Manfred-Fuchs-Platz 2-4, 28359 Bremen, Germany, daria.stepanova@skolkovotech.ru

Abstract

The continuing Machine Learning (ML) revolution indubitably has had a significant positive impact on the analysis of downlinked satellite data. Other aspects of the Earth Observation industry, despite being less susceptible to widespread application of Machine Learning, are also following this trend. These applications, actual use cases, possible prospects and difficulties, as well as anticipated research gaps, are the focus of this review of Machine Learning applied to Earth Observation Operations. A wide range of topics are covered, including mission planning, fault diagnosis, fault prognosis and fault repair, optimization of telecommunications, enhanced GNC, on-board image processing, and the use of Machine Learning models on platforms with constrained compute and power capabilities, as well as recommendations in the respective areas of research. The review tackles all on-board and off-board applications of machine learning to Earth Observation with one notable exception: it omits all post-processing of payload data on the ground, a topic that has been studied extensively by past authors. In addition, this review article discusses the standardization of Machine Learning (i.e., Guidelines and Roadmaps), as well as the challenges and recommendations in Earth Observation operations for the purpose of building better space missions.

Keywords: *Artificial Intelligence; Astrionics; Earth Observation; Edge Computing; Machine Learning; Neural Network; Remote Sensing; State-of-the-art*

Acronyms / Abbreviations

Artificial Intelligence (**AI**); Machine Learning (**ML**); Deep Learning (**DL**); Fault Detection, Isolation and Recovery (**FDIR**); Guidance, Navigation and Control (**GNC**); Neural Network (**NN**); Convolutional Neural Network (**CNN**); Deep Neural Network (**DNN**); Artificial Neural Network (**ANN**); Binarized Neural Network (**BNN**); Bayesian Network (**BN**); Dynamic Bayesian Network (**DBN**); National Aeronautics and Space Administration (**NASA**); European Space Agency (**ESA**); On-Board Computer (**OBC**); Earth Observation (**EO**); Random Decision Forest (**RDF**); Bayesian Thresholding (**BT**); Support Vector Machine (**SVM**); Commercial off-the-shelf (**COTS**); Size, Weight and Power (**SWaP**); Light Detection and Ranging (**LIDAR**); System on a Chip (**SoC**); False Positives (**FP**); Consultative Committee for Space Data Systems (**CCSDS**); Global Navigation Satellite System (**GNSS**); Global Positioning System (**GPS**); Proportional - Integral (**PI**); Proportional - Integral - Derivative (**PID**); Attitude Orbital Determination System (**AODS**); Reinforcement Learning (**RL**); Extended Kalman Filter (**EKF**); Random Forest (**RF**); Attitude and Orbit Control System (**AOCS**); k-Nearest Neighbour (**k-NN**); Self-Organizing Map (**SOM**); On Orbit Servicing (**OOS**); Anomaly Resolution and Prognostic Health Management for Autonomy (**ARPHA**); Density-Based Spatial Clustering of Applications with Noise (**DBSCAN**); Electrical Power System (**EPS**); Software and Sensor Health Management (**SSHM**); Regularized Discriminant Analysis (**RDA**); Adaptive Regularization of Weight Vector (**AROW**); Soft Confidence-Weighted (**SCW**); Centre national d'études spatiales (The National Centre for Space Studies) (**CNES**); European Space Operations Centre (**ESOC**); Radial Basis Function (**RBF**); Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center) (**DLR**); One-Class Support Vector Machine (**OC-SVM**); Normal Gaussian Herding (**NHERD**); Thermal EMission Imaging System (**THEMIS**); Intelligent Payload EXperiment (**IPEX**); Hyperspectral Infrared Imager (**HyspIRI**); Moderate-Resolution Imaging Spectroradiometer (**MODIS**); Peak Signal to Noise Ratio (**PSNR**); Structural Similarity Index (**SSIM**); Field Programmable Gate Array (**FPGA**); Context-Based, Adaptive, Lossless Image Codec (**CALIC**); Integral Wavelet Transform (**IWT**); Peano-Hilbert (**PH**); Learning Vector Quantization (**LVQ**); TensorFlow (**TF**); Synthetic Aperture Radar (**SAR**); Ratio of Exponential Weighted Average (**ROEWA**); Joint Photographic Experts Group (**JPEG**); Space Test Program-Houston-5-Cubesat Service protocol (**STP-H5-CSP**); Neural Architecture Structure (**NAS**); Central Processing Unit (**CPU**); Graphics Processing Unit (**GPU**); Visual Processing Unit (**VPU**); Time Processing Unit (**TPU**); Trained Ternary Quantization (**TTQ**); Radiation Tolerant (**RT**); Mobile Neural Architecture Search (**MNAS**); Knowledge Transfer (**KT**); Knowledge Distillation (**KD**); Cubesat Service Protocol (**CSP**); SpaceBorne Computer (**SBC**); Modified

National Institute of Standards and Technology database (**MNIST**); National Information Security Standardization Technical Committee (**NISSTC**); Deutsches Institut für Normung (German Institute of Standardization) (**DIN**); Deutschen Kommission Elektrotechnik Elektronik Informationstechnik (German Commission for Electrical, Electronic and Information Technologies) (**DKE**); European Union (**EU**); European Commission (**EC**); Small and Medium Enterprises (**SMEs**); DEpendable and Expandable Learning (**DEEL**); Earth Observation Systems' Data Information Systems (**EOSDIS**); Space Generation Advisory Council (**SGAC**); Small Satellite Project Group (**SSPG**); Deep Reinforcement Learning (**DRL**); Geographic Information Systems (**GIS**); Explainable AI (**XAI**).

1. Introduction

Earth Observation (EO) satellites have allowed us to look at our planet at a scale previously unattainable to humankind. From the vantage point of space, it becomes easier to monitor everything about our lives on a very large scale, such as our impact on the planet's ecology (Guo et al., 2017) and extent of specific facilities all around the world (Pan et al., 2021). This capability has been and continues to be invaluable to understanding the world around us and enforcing regulations vital to the well-being of people all over the globe.

However, as access to space becomes ever more affordable, EO assets multiply at an increasingly faster pace (Belward and Skøien, 2015). Moreover, EO Operations - the sequence of activities that take place in managing an EO spacecraft from its launch to its demise - keep growing in number and complexity as new assets are put into orbit. These trends could soon lead to a situation where available work-power becomes a limiting factor in the deployment of EO systems. Orchestrating these operations is, at its core, a control and data processing problem - from taking in and analyzing large volumes of telemetry from all EO platforms to taking into account their complex dynamics and evolving mission profiles when utilizing them.

Artificial Intelligence (AI) is becoming more prevalent in our daily lives, whether it is in the form of personalized newsfeeds, shopping online or streaming movie recommendations, or even mapping tools that help us avoid traffic jams. On a larger scale, AI is already having a significant impact on healthcare, banking, agriculture, and a variety of other industries, and its impact is expected to grow quickly in the coming years. Machine Learning (ML), a subset of Artificial Intelligence (AI) as shown in Figure 1 wherein machines learn from data, has been used in a variety of space-related applications. In our review, we considered that ML is a subfield of AI for clarification. Deep learning (DL) is a subfield of machine learning.

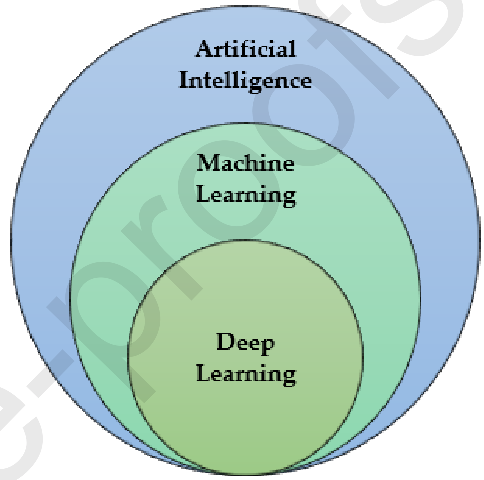

Figure 1: AI, ML, DL relationship (Zhang et al., 2021).

Human analysts may miss patterns and trends hidden within massive amounts of data, but ML can find them.ML, on the other hand, can uncover patterns and trends hidden inside massive amounts of data that are invisible to human researchers. Modern Earth Observation systems collect a massive amount of data from a variety of sensors with varying temporal, spatial, and spectral resolutions. Because of its complexity, it necessitates the use of innovative procedures and methods to extract useful information. Figure 2 represents a typical machine learning process.

.1.1. *Machine learning for Earth Observation*

ML has taken the data processing world by storm, with one success story after another. From object detection and classification (Krizhevsky et al., 2012) to natural language processing (Wang, 2021) and nonlinear control (Mnih et al., 2013), the capacity of these algorithms to solve different types of problems has been nothing short of awe-inspiring.

For the purposes of the present review, we define ML algorithms as those whose performance critically depends on and generally improves with exposure to real-world data of the problem to be solved.

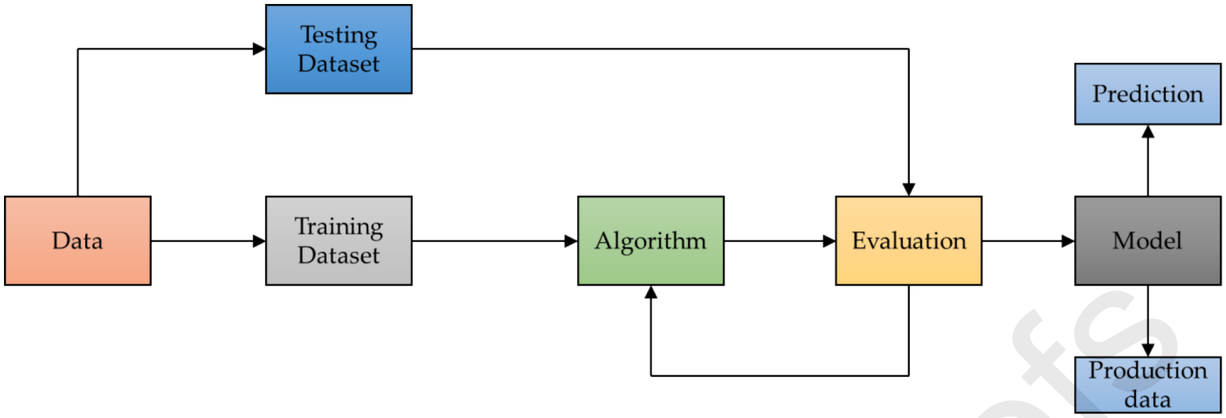

Figure 2: Machine Learning workflow (Pant, 2019).

So far, these techniques have concentrated mostly on the analysis of downlinked imagery due to the larger availability of computing power and ease of deployment relative to on-board applications, as well as its relatedness to computer vision, one of the traditional strong suits of ML.

But applications to other aspects of operations are now starting to surface. The present review explores the contexts for which these applications have been proposed or in which they have been applied, exposes the possibilities that they open up and risks that must be avoided, and illustrates gaps in research that we believe should be addressed by the Earth Observation community. As shown in Figure 3, ML will enhance space exploration operations in a variety of ways, particularly for Earth observation missions.

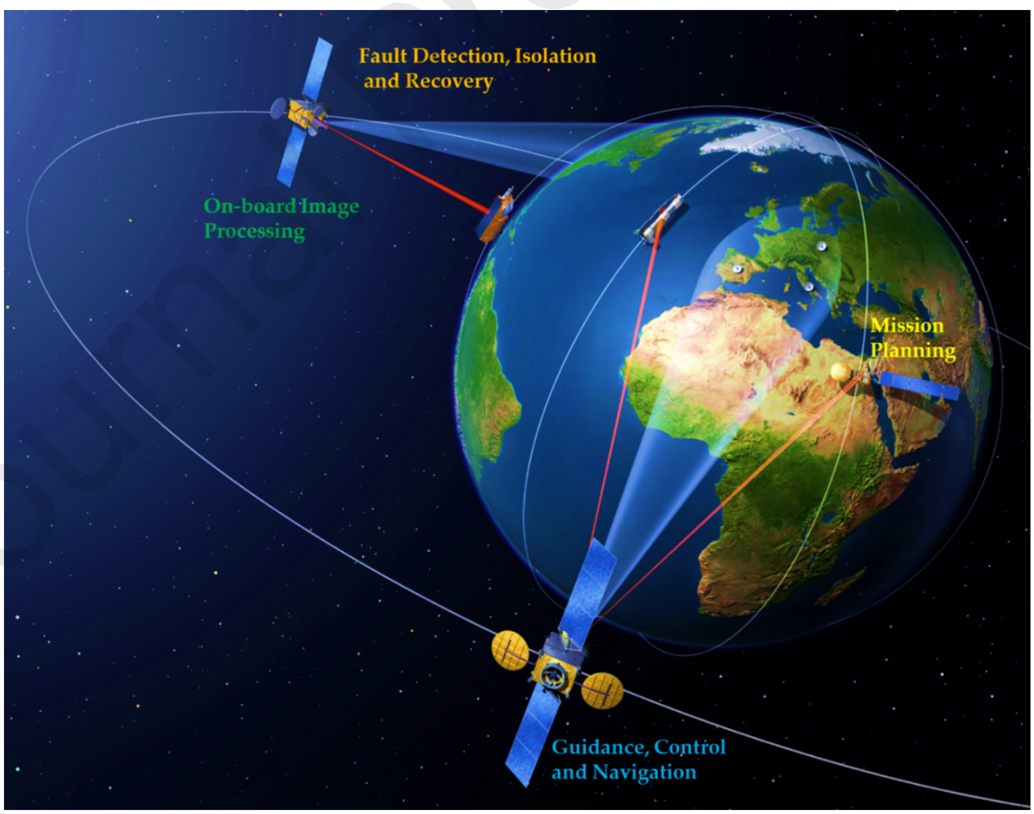

Figure 3: Potential application of ML for earth observation mission.

As illustrated in Figure 3, the manuscript discusses the state-of-the-art as well as the future prospects of ML in Mission Planning (section 2), GNC (section 3), FDIR (section 4), and on-board image processing (section 5).

In section 6, we also discussed the useful aspect of using ML models in EO operations. We examine how operators might make the most of their limited on-board resources by properly optimizing the usage of ML model resources, describing a variety of software and hardware solutions geared to that end. In Section 7, we also discuss recent initiatives within the space sector to standardize and guide the deployment of ML models, as well as the kinds of considerations a designer must make in order to avoid frequent errors with this technology.

Our objective is to give EO Operators a thorough, if not exhaustive, assessment of the current situation with regard to ML applications in their area of expertise. The review connects EO operators and proponents of ML algorithms for EO Operations problems in an effort to spark discussion and stimulate additional application suggestions and demonstrations.

With one important exception, the review covers all on-board and off-board applications of ML to EO but leaves out all post-processing of payload data on the ground, a subject that has been intensively researched by other researchers.

The optimization of tracking, telemetry, and command is another subject we pass over. Despite the fact that this was initially our intention, we discovered an outstanding and current review by Fourati and Alouini (Fourati and Alouini, 2021). We invite interested readers to check out the excellent paper rather than pointlessly duplicating their work.

This article has been reviewed and updated in comparison to the conference paper presented at IAC in 2021. This research, which is highly needed in present space industry, was conducted by a group of volunteers from the Small Satellite Project Group (SSPG) of SGAC. SGAC is a non-profit, non-governmental organization with over 16,000 members dedicated to the peaceful uses of space. There are over a hundred active volunteers, in addition to eleven project organizations, including the SSPG. The SSPG focuses on how small satellites are utilized in the space industry and how they can assist humanity in realizing space's full potential.

2. Machine Learning in Earth Observation Mission Planning

There are many constraints to mission planning. Some relate to the target area: It needs to be under the satellite and illuminated by the sun at capture time (orbit and time-dependent); clouds are to be avoided (weather dependent); the requester may set a deadline and/or a priority. Others relate to the satellite, such as limited memory capacity; limited transmission capability; reduced communication opportunities with the ground antennas; multiple sensors to choose from; and limited maneuverability to skew the observation angle and reach areas not directly flown over.

All of these parameters make optimal scheduling of observations a highly combinational problem for a mission that supports multiple independent requests, and it is even more complex when they are accomplished by a constellation of satellites. ML proposes a series of algorithms that may find better solutions than non-learning algorithms or do so more efficiently.

There are many different formulations of the observation scheduling problem, taking into account different subsets of the constraints presented in the previous paragraphs, adapted for different types of missions and ground segments.

2.1. Classical approaches

Non-ML algorithms to the satellite scheduling problems can be classified into two categories: Exact and Heuristic methods (X. Wang et al., 2021).

Exact methods typically consist of a combination of branch and bound methods and mixed-integer linear programming. These methods are computationally costly and can become intractable for moderately sized constellations.

Heuristic methods use an approximated rule to guide the construction of a solution. Greedy algorithms construct a solution by gradually choosing the best action at every decision step according to some metric, without regard as to how the overall sequence of decisions plays out. Other heuristic methods include backtracking through constraint programming and search algorithms. Other forms of search include hill-climbing or squeaky-wheel optimization, where the geometry of the optimization functions is exploited to accelerate the search process. Globus et al. (Globus et al., 2003) compare multiple algorithms such as genetic, simulated annealing, squeaky wheel and hill-climbing on a problem with one or two satellites.

Evolutionary or genetic algorithms simulate processes akin to biological evolution to optimize candidate solutions according to a hand-crafted fitness function. Mansour et al. (Mansour and Dessouky, 2010) studied the performances of a genetic algorithm for a single satellite with limited memory and multiple instruments and imaging modes. Li et al. (Li et al., 2014) explore genetic algorithms in order to provide scheduling in real-time, optimizing the transmission path towards the user.

Simulated annealing imitates the annealing processes found in metals exposed to high temperatures, and it forms the basis for another branch of heuristic algorithms. The simulated annealing seems to provide better results, confirmed by Globus et al. (Globus et al., 2003) in a complete multi-satellite formulation of the problem, including satellite agility and priorities.

Lastly, multi-agent systems simulate interactions between simple agents representing part of the systems to determine an optimal policy. Bonnet et al. (Bonnet et al., 2015) use a self-adaptive multi-agent system for real-time and robust adaptation of a multi-satellite problem, including request priorities.

2.2. *ML-based approaches*

ML-based approaches can exploit the statistical distribution of typical problem settings to accelerate the finding of good solutions to the mission planning problem.

Wang et al. (X. Wang et al., 2021) present a comprehensive review of publications on the agile observation scheduling problem, including ML and non-ML approaches. The authors classify approaches along multiple axes such as time continuous and discrete-time model, type of solving method and also other features such as autonomy, and multi-objective profit function.

Neural Networks (NNs) are explored by Wang et al. (Wang et al., 2019) in order to provide immediate results for a multi-satellite mission using Deep Reinforcement Learning (DRL). Peng et al. (Peng et al., 2018) apply recursive NNs in a sequential decision-making process in order to achieve low scheduling computation time and high performance when compared to a deterministic resolution. Recursive NNs allow the model to condition current decisions on past inputs, instead of depending exclusively on the present inputs to the system, providing the model with a sort of memory. We have not found any applications of Transformers to this problem, a sequence modeling technique from the deep learning research field that has shown excellent results in other sequential tasks like language modeling and even in image processing tasks.

Neuroevolutionary techniques combine the advantages of neural models and evolutionary algorithms. Du et al. (Du et al., 2020) leverage a prediction model trained by a Cooperative Neuro-Evolution of Augmenting Topologies algorithm in order to filter tasks to be scheduled according to the probability to be fulfilled before scheduling using genetic algorithms. DRL uses NNs as function approximators to approximate hard to determine functions in dynamic programming. This has enabled groundbreaking achievements in other control and scheduling problems like playing Go or automated driving. Despite its potential, it has not been extensively applied to this problem set. Liu (Liu, 2020) applies Proximal Policy Optimization, a method of the DRL literature, to mission planning for a single satellite. Unfortunately, they do not compare performance to other methods or extend it to a multi-satellite setting.

Yuchen et al. offer a unique online strategy that combines a Q-network with a pruning technique to address the observation sequence planning problem. The proposed scheme's goal is to generate an observation sequence based on the Q-learning heuristic rule and increase the neural network's efficiency in optimization. A Q-network-based mission-planning algorithm for the operation of EO satellite is shown in Figure 4. It shows the suggested algorithm's overall workflow (Liu et al., 2021).

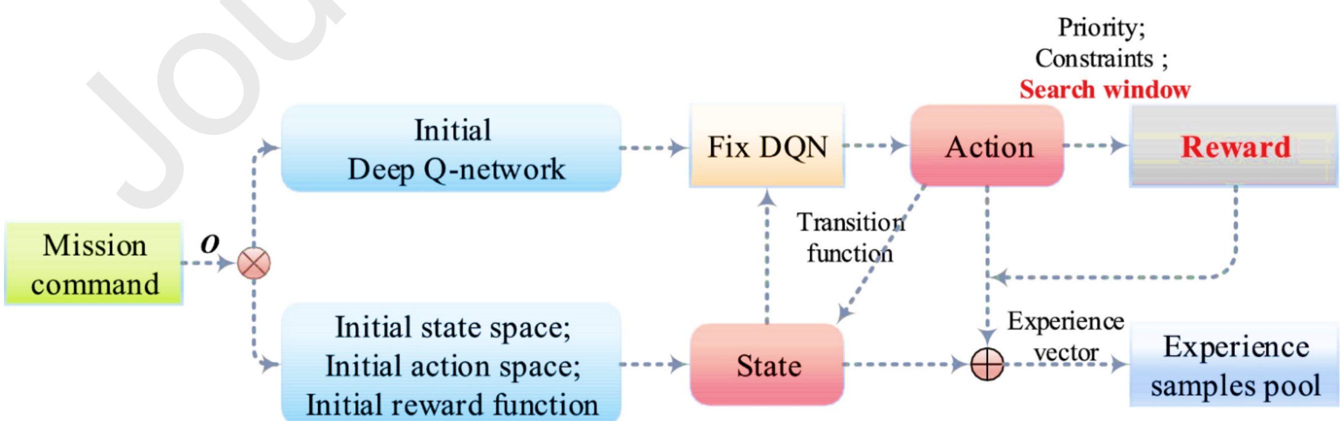

Figure 4: ML-based mission planning algorithm (Liu et al., 2021).

Hadj-Salah et al. (Hadj-Salah et al., 2020, 2019) explore the application of Actor-Critic (A2C), a DRL algorithm, to the mission planning problem. They compare it to random planning and a planning heuristic that compromises between greedy and long-term planning. Their models are trained in a simulated mission planning environment and then executed in a real test scenario. Their long-term version of A2C shows better performance than the heuristic algorithm. In their later publication, they augment the training process with techniques from the domain randomization and transfer learning literature, meant to increase robustness to the gap experienced when passing from the simulated training scenario to the real validation scenario.

2.3. *Recommendations*

Mission planning is a very rich problem that has been explored for many years using machine learning amongst other solutions.

Comparing the performances of algorithms presented in different papers is not a suitable path because each presents its own definition of the problem, with a unique set of constraints, different mission characteristics, variable satellite capabilities and potentially incompatible metrics. For instance, a lot of schedulers take into account satellite memory, limiting the number of observations until a ground station is visible, but few of them also make sure the ground station is available for communication with the satellite and not busy communicating with another one of the constellations. Song et al. (Song et al., 2020) introduce a framework in order to facilitate future comparisons but additional work on model standardization is needed before results from different studies can be compared.

We observe a shifting trend in algorithms applied to this problem over the years from genetic or annealing to ML approaches such as NNs. Unfortunately, we found no sources comparing the performances of genetic and ML-based schedulers on a single problem formulation.

Standardizing project formulations, constraints set, and optimization metrics seem to be a necessary step for sustainable collaborative research in this field. Relying on Consultative Committee for Space Data Systems (CCSDS) published standards and models could be a first step in the direction of a unified approach.

3. Machine Learning in Earth Observation Guidance, Navigation and Control

GNC, describe the set of operations needed to move a satellite platform or any other vehicle. The guidance relates to planning paths from a current state to the desired state. Navigation is the determination of the present state. Control is the correct use of spacecraft actuators, such as an engine, to execute the desired plan.

3.1. *Classical approaches*

Two main tasks need to be achieved by a GNC system: determination of the current state, which is an estimation task, and use of the spacecraft's actuators to go from the current state to the desired state, which is a control task.

Spacecraft control is typically subdivided into at least two different granularity levels, guidance and control, where guidance is the high-level control of the spacecraft from a current dynamic state to a future one. A guidance module may output the sequence of feasible dynamic states necessary to achieve a new orbit from the required orbit. EO Satellite maneuvers are often planned and optimized on the ground, and the onboard guidance modules are minimal. For the control task, a number of control schemes are used, most notably controllers from the robust control literature such as $H\infty$ controllers.

As for navigation, spacecraft state is typically determined via variants of the Kalman Filter (KF), such as the Extended Kalman Filter (EKF)or the Unscented Kalman Filter (UKF). These methods are model based, that is, they depend on an explicit model of spacecraft dynamics for their calculations.

Fuzzy controllers have been proposed as a possible improvement to the classical approach. The literature contains several works where GNC and Attitude and Orbit Control System (AOCS) controllers based on fuzzy logic are compared to their traditional counterparts. For instance, Wu et al. (Wu et al., 2001) studied the fuzzy logic controller with the X-38 re-entry vehicle. ESA also investigated the usage of fuzzy logic controllers to carry out Geostationary Equatorial Orbit (GEO) rendezvous autonomously (Ortega, 1995) to aid in in-orbit manufacturing. As another example, in (Cheng et al., 2009), a simulation of ROCSAT-1 / FORMOSAT-1's attitude controller is carried out, where the classical setup of a Proportional - Integral (PI) pitch axis controller and Proportional - Integral - Derivative (PID) roll/yaw axis controller is replaced with two fuzzy controllers initially, and a single consolidated fuzzy controller afterwards, yielding considerable improvements against interference as well as a lower steady-state error.

Nevertheless, despite the body of research backing up their effectiveness, there is no widespread use of fuzzy logic GNC controllers for space missions.

3.2. ML-based approaches

Izzo et al. (Izzo et al., 2018) present a survey of Artificial Intelligence applied to GNC which, is not focused on EO applications, can nonetheless be useful to practitioners. The survey contains a section focusing on ML approaches, on top of other AI approaches such as evolutionary algorithms.

In another publication (Izzo and Öztürk, 2021), Izzo and Öztürk leveraged DRL to plan near-optimal real-time computation of low-thrust transfers. They also suggest a new method to generate training data for such problem settings. Although originally designed for Earth-Venus transfers, their solution is applicable to all low-thrust transfers, but the data generation algorithm and optimality comparisons are problem-dependent.

ML excels in problems where no structured pre-existing model can be exploited. That is not the case for the GNC problem, where the general form of the dynamics governing spacecraft are well known and solvable. It is, however, the case for visual-based GNC, as no model exists for relating camera inputs to dynamic state or control actions. For this reason, much research on ML-powered GNC has focused on visual-based GNC (Frédéric Férésin et al., 2021) for autonomous rendezvous. This, however, is not directly relevant to the EO Operations community, who are unlikely to engage in autonomous docking as providers.

An interesting streak of research looks into applications of ML to processing visual navigation sensors, particularly Earth and Sun sensors and star trackers. Koizumi et al. (Koizumi et al., 2018) present a DL-powered Earth sensor capable of determining the attitude of the spacecraft by processing the images captured by a Commercial-Off-The-Shelf (COTS) camera. It runs a real-time image processing algorithm to extract features into the images separating them into distinct feature sets using DL techniques. The features sets are then compared to the preloaded data sets to determine the position of the spacecraft relative to Earth in the 3D plane. The primary advantage of the system is the use of a COTS component and a single board computer.

Another research thread explores the combination of ML techniques and fuzzy controllers (Kim et al., 2016). Classical fuzzy controllers rely on manually set parameters that define behavior. This research thread attempts to leverage ML techniques to learn the optimal value for these parameters from a training dataset. These have the advantage of interpretability - their reliance on explicitly (if fuzzily enforced) rules means that they remain grounded on human-interpretable system models. Joghataie' PhD. thesis (Joghataie, 1994) suggests the development of a neuro-fuzzy controller, wherein the tuning of the fuzzy logic is performed automatically by using neural networks in a hybrid approach. Azarbad (Azarbad et al., 2014) suggests a model applied to Global Positioning System (GPS) systems that outperform the classical fuzzy controller. A simulation study on MATLAB was done by Baranwal et al. in (Baranwal et al., 2018) , comparing the performance of a PID controller and a fuzzy PID controller for a student satellite team. The EKF-based fuzzy controller outperformed the classical controller. The study was done on a 3U CubeSat. Further research can be done comparing these controllers with ML-based approaches. We have been unable to find a comparison between the three types of controllers, i.e., neuro-based controller, fuzzy controller and a hybrid model, as implementation details in different studies differ, complicating their comparison.

Wang et al. (Wang et al., 2019) have developed a DL framework that stabilizes the spacecraft using a real-time torque control. It is initially trained in a simulation environment, enabling it to learn the required torque output and extrapolate it for unknown disturbances. It performs better than a conventional PID controller, as it can correct the attitude after unknown disturbance rather than repeatable corrections. A similar system is proposed by Yadava et al. (Yadava et al., 2018). They propose an Attitude Orbital Determination System (AODS) system that determines the position of the spacecraft, taking inputs from the magnetometers (magnetic vectors) and sun sensor (sun vector) along with GPS data (position and velocity vector), and determines the ideal attitude depending on the position using a neural network. The required torque calculations are made and sent to the Reinforcement Learning (RL)-based controller to make the required adjustments. The system performs better than classical PID controllers as it consumes less computation power for subsequent cycles as the algorithm learns.

3.3. Recommendations

Most ML for GNC applications in the space sector seem to have been explored in the context of space logistics and space exploration rather than Earth Observation. Although guidance and control for EO platforms are simple compared to these applications, we believe there is a potential to adopt some of these technologies.

Attitude determination is a domain where EO operations have high requirements. We believe that vision-based processing applied to this area is just getting started, and that use of more refined neural architectures could enable improvements in performance or resource consumption compared to current approaches.

4. Machine Learning in Earth Observation Fault Detection, Isolation, and Recovery

Satellites performing EO tasks have stringent requirements in terms of accuracy, continuity and stability of payload operations. To this end, Fault Detection, Isolation and Recovery (FDIR) is focused on developing and improving tools to guarantee and maintain reliable spacecraft operations. FDIR describes a set of engineering disciplines focused on safeguarding and maintaining the spacecraft in nominal operating conditions. The target of these disciplines is represented by faults, irregular occurrences and processes with the potential to disrupt the mission up to the point of failure.

ML can be an extremely powerful tool for FDIR. Indeed, the core capability provided by ML is pattern detection. Therefore, ML can be used both to detect anomalies in the telemetry or outputs from any subsystem (diagnosis) and identify signs indicating an incipient fault (prognosis). This section presents relevant ML literature for four significant sub-topics: fault detection, fault diagnosis, recovery, and fault avoidance

4.1. Fault Detection

Failure detection deals with identifying the presence of faults and their rates of occurrence.

4.1.1. Classical approaches

In classical approaches, the recognition of failures is mainly based on constant thresholds and fixed logic diagrams defined during the design process. (Wertz and Larson, 1999) One of the key issues with classical fault detection is model brittleness. As fault detection schemes are based on hardcoded thresholds, these models are easily disrupted by noise and deviations from theoretical assumptions.

4.1.2. ML-based approaches

An example of an ML-based solution to the issue of model brittleness can be found in a paper by Jaekel et al. (Jaekel and Scholz, 2015). This work uses Self-Organizing Maps (SOM), an unsupervised variant of Artificial Neural Networks (ANNs), for the detection of failures in dexterous manipulators for On-Orbit Servicing (OOS). SOM manages to adapt to the idiosyncrasies of incoming data in a simulated environment and thus show increased robustness to input variations with respect to traditional methods. They can also deal with uncertainties and noise in values. A dexterous manipulator on a maintenance satellite captures a client spacecraft having 7 degrees of freedom. They inject sensor failures, including sensor outage and drift, during arm operations and the results show that SOMs are a robust approach as temporary fluctuations in the sensor, outliers and peaks do not unnecessarily stop the current operation. But the computational load is relatively high and needs to be optimized to reduce system reaction time. The authors suggest improving the precision and speed of the method by adding more information from redundant sensors. Ranasinghe et al. provides a comprehensive analysis of FDIR (Ranasinghe et al., 2022). Fuertes et al. (Fuertes et al., 2018) discuss ML-based fault detection using NOSTRADAMUS, an algorithm developed by the Centre National des Études Spatiales (CNES). NOSTRADAMUS uses a One-Class - Support Vector Machine (OC-SVM), a common algorithm used to detect outliers, to detect the presence of an anomaly in telemetry data. NOSTRADAMUS runs on the ground segment, analyzing telemetry as it is downlinked from the satellite. The performance of NOSTRADAMUS is compared to algorithms inspired by Novelty Detection (ESOC), Project Sybil (Ivano Verzola et al., 2016), and ATHMoS (DLR) (O'Meara et al., 2016). NOSTRADAMUS is the best option because it has a 100% detection rate and the minimum false alarm rate (5 percent). The Novelty-inspired algorithms show the best performance of false alarm reduction, with 85 percent of valid detections and fewer than 1% of false alerts.

CNES is working on an on-board version of this algorithm, as well as on extensions to the ground-based variant for processing of multiple telemetry variables based on dictionary learning approaches (Pilastre, 2020). In conversations during their collaboration with this project, CNES teams signaled that explainability was a crucial aspect of any technique. Being able to understand the features of input data that signal a fault lets the operational teams understand the context of their satellite and know which actions must be taken to remedy the situation - this is comparable in value to being able to detect the anomaly in the first place.

Project Sybil is a collaborative effort by DLR's Columbus Flight Control team, ESA's Advanced Mission Concept Section, and Ludwig Maximilians Universitat to apply an outlier identification algorithm to the Columbus telemetry database. After data segmentation and computation of its respective characteristics, it uses the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) technique to preprocess the data. It is an unsupervised clustering method that divides data into a variable number of clusters based on their relative distances. Following this grouping, clusters with less than 5% of the population data are discarded on the assumption that they may indicate a non-nominal working mode in the learning dataset. Project Sybill allows for higher mission performance by reducing downtime caused by onboard system failures.

4.2. Fault Diagnosis

Fault detection identifies the presence of faults and performance degradation, while fault diagnosis identifies the root causes of these events.

4.2.1. Classical approaches

Though traditionally, fault diagnosis has been achieved by human operators at the ground through comparison with hardcoded hand-tuned thresholds. It is difficult to deal with large amounts of data using this approach. Iverson et al. (Iverson, 2008) point out that for efficient utilization of the data, there is a need for an autonomous approach that eliminates the necessity of human experts for diagnosis.

4.2.2. ML-based approaches

Ricks et al. (Ricks, 2021) examine fault detection and identification for a satellite Electrical Power System (EPS) testbed using BNs compiled to arithmetic circuits. BNs can be used to model partial knowledge and uncertainty by identifying the system state based on probabilistic relationships between a set of system variables at a certain instance in time (Meß, 2019). The proposed methods work for complex systems exhibiting both continuous and discrete behavior. The discussed techniques can handle abrupt continuous faults particularly well, which often pose problems. For example, a nominal value region is not enough to detect offset faults if they are small enough - the paper uses cumulative sums to deal with these. Additionally, “stuck” faults may be difficult to detect in low-noise conditions since fluctuations might be infrequent. The authors employ a tunable time interval which will mark the sensor as working abnormally after it expires without the readings having made any change. Different types of nodes, modeling different behaviours, are grouped to defined sensors and components, which in turn are assembled to create the entire EPS functional FDIR structure. BNs have also been used by Schumann et al. (Schumann et al., 2011) to detect onboard failures and perform diagnoses. A Software and Sensor Health Management (SSHM) system is developed for a simple GNC structure of a small satellite using BNs that collect data from hardware sensors, software quality signals, software status signals and data from the operating system in order to determine whether any failures exist, what the most likely causes are, and to provide a statistically sound quality measure of the diagnose. The developed SSHM system requires no modification to the satellite subsystems for which it performs FDIR - it just uses the sensor data outputs. That way, model-level and code-level Verification & Validation can be performed independently on the SSHM system to certify that the rate of false positives and false negatives is below a selected threshold. This SSHM, applied to a simple GNC system, was able to detect and diagnose both hardware and software problems successfully. Nevertheless, it remains a simplistic case and more research into hierarchical SSHM systems is required in order to apply them to large-scale BNs. The approach can further be extended to failures that are not modeled and unexpected and due to arising behavior.

Although not specifically related to space systems, Liu et al. (Liu et al., 2018) reviews the existing techniques for ML-based fault diagnosis in rotating machinery. In general, it presents useful research and conclusions which we consider can be applied to reaction wheels in the AOCS subsystem of spacecraft. K-Nearest Neighbor (k-NN) is the simplest method reviewed, which exhibits ease of implementation but necessitates careful fine-tuning and large computation and storage space. The authors cite BNs’ strong prior assumptions as the biggest shortcoming of this family of algorithms while mentioning as main advantages that it possesses a clear physical explanation of how it detects faults and its reduced storage space requirement. Support Vector Machine (SVM) is also reviewed, and its high-dimension accuracy is highlighted, even if the physical meaning is obscured, unlike with the previous two techniques. Finally, DL techniques have the potential to learn from data up to a degree of complexity much higher than any of the other techniques without the need for a manually crafted feature extractor. However, the main drawback of this approach is the need for large samples in order to train the network, which is difficult to obtain unless the spacecraft is a new iteration of previously flown models for which data already exists. If the satellite is a one-off, this can only be obtained in an approximate manner by creating a simulation environment. The authors underline that future ML-based fault diagnosis methods should not be purely data-driven but should consider possible failure mechanisms, system models and prior knowledge in general to increase diagnostic performance.

Voss (Voss, 2019) explores the use of DL for fault detection and isolation in a simulation environment. A NN is developed, trained offline and tested to detect and isolate single faults in the reaction wheels, GPS, star tracker and magnetometer subsystems, as well as two simultaneous faults. A case study with PROBA-V mission parameters is also performed for the AOCS subsystem only. The implemented system yielded mixed results: while some subsystems have a near-perfect performance, the network fared poorly regarding others, namely misalignment faults. Also, fault isolation was much more reliable than fault detection. On top of that, a large dataset is required for this system to work, so creating a simulation environment is mandatory, especially for one-off spacecraft, to acquire enough data for adequate network training. This study also assumes there is enough electrical and computing power available on the satellite to run this deep-learning-based solution. We overview techniques for reducing resource consumption of deep neural models and other techniques in section 6.

4.3. Recovery

In FDIR, recovery entails reconfiguring the problematic element and/or the entire spacecraft to restore normal system behaviour (Jaekel and Scholz, 2015).

4.3.1. Classical approaches

Traditional FDIR is able to respond to predefined events by selecting a recovery path from the available set of options. However, the status of the system and its environment can exhibit various kinds of uncertain behavior due to their dependence on the internal subsystem, component reliability factors, external environment factors (e.g., illumination conditions, thermal, radiation) and on system-environment interactions (e.g., resource utilization profiles, stress factors, degradation profiles) (Meß, 2019). Due to these uncertainties, the system and its environment cannot be completely observed by traditional FDIR concepts that pose limitations to autonomous isolation and recovery (Meß, 2019). For example, Mars Express lost six months of operational hours due to a non-resolvable memory problem that forced it into safe mode repeatedly (Jaekel and Scholz, 2015).

4.3.2. ML-based approaches

Raiteri et al. (Codetta-Raiteri and Portinale, 2015) discuss the use of Dynamic Bayesian Networks (DBNs) to address issues like partial observability, uncertain system evolution and system-environment interaction, as well as the prediction and mitigation of imminent failures. The BNs do not model the relationship between variables at previous points in time. DBNs are an extension to BNs that refer to past values of certain variables to express dynamic aspects of the system over discrete time (Meß, 2019). The approach is applied by Raiteri et al. (Codetta-Raiteri and Portinale, 2015) onto the power subsystem of a simulated ExoMars rover, by simulating different failure scenarios. The DBNs can infer whether the system is currently in a normal, anomalous or failed state. On detection of a failure, a suitable recovery plan is suggested. A preventive recovery plan may be proposed in case an anomaly is inferred. The FDIR presented in this paper also has the capability of performing a prognostic state estimation that can also be used for preventive recovery. The proposed approach has been implemented in an on-board software architecture called Anomaly Resolution and Prognostic Health Management for Autonomy (ARPHA). The results show that DBNs are suitable for failure situations requiring autonomous (preventive and reactive) recovery.

AIKO Technologies have developed a software library, MiRAGE, that can enable the spacecraft to make autonomous decisions for processing telemetry and payload. The library is meant to be installed on the satellites to enable functionalities such as event detection, predictive maintenance and autonomous re-planning.

4.4. Fault Avoidance

Fault avoidance methods are concerned with preventing the occurrence of faults.

4.4.1. Classical approaches

FDIR in past missions worked under the notion that a fault is detected and then the algorithm will react, according to predefined scenarios.(Jalilian et al., 2017; Olive, 2010). Regarding ML-based models, one of the bottlenecks to having an on-board failure avoidance system is that the models are trained on the ground with limited data that does not represent actual behavior in space. This gives rise to the requirement of real-time access to the data, which can be used to represent multiple onboard scenarios, and closely represents spacecraft behavior during the mission.

4.4.2. ML-based approaches

Especially notable in the context of ML-enabled fault avoidance is the work of Labrèche et al.(Georges et al., 2021) discussing the OrbitAI experiment onboard the OPS-SAT spacecraft. OPS-SAT is a special ESA satellite deployed with the scope of being a testbench for novel software technologies in orbit. OrbitAI uses ML techniques to obtain intelligent FDIR algorithms enabling the onboard camera to avoid direct exposure to sunlight. Interestingly the ML model used is trained on-board, rather than offline. The model is trained with five training algorithms tested of those natively provided in the MochiMochi library (olanleed, 2021) for online ML training: Adam, RDA, AROW, SCW, and NHERD. When using the figure of merit of balanced accuracy, only one model appears to achieve values significantly different from 0.5: the AROW algorithm in three-dimensional input space.

4.5. Recommendations

FDIR innovation has been applied mainly to deep space missions, which need a higher degree of autonomy due to their long communication delays inherent to the long distances traveled. However, the analysis of the literature suggests that ML in EO FDIR has promising prospects. The approach can be extended to diagnose failures that are not modeled, unexpected and due to arising behavior, which offers a great advantage in overcoming the model brittleness issues of traditional FDIR. ML-based fault detection and diagnosis solutions can be integrated alongside the traditional FDIR of the satellite. But ML-based recovery is virtually unexplored, and much research is needed in this domain. The majority of the work in this field concerns BNs, while other research avenues remain largely unexplored, such as ANNs and DL. As it would be shown and discussed in Section 6, power and computational resources remain a big concern for ML-based FDIR, especially for small satellites. The benefits of ML-based FDIR can be further researched to be implemented in future EO satellites to perform FDIR on the AOCS subsystem, GNC, On-Board Data Handling, Power subsystem, and detection of faulty sensors.

5. Machine Learning in Earth Observation On-board Image Processing

5.1. On-board Image Processing

Clouds cover 66% of the Earth's surface and are an obstacle when observing the Earth's surface in certain wavelengths such as visible light. Removal of clouds from satellite images is an important preprocessing phase for most of the applications in remote sensing.

Researchers have explored various forms of Cloud detection like "Cloud / No cloud", "Snow / Cloud", and "Thin Cloud / Thick Cloud", using various approaches of ML and classical algorithms (Mahajan and Fataniya, 2020). Cloud detection/filtering can be used alongside novelty detection. Novelty detection is to detect unexpected features and it is especially important while looking into new environments.

Good cloud detection algorithms are necessary to optimize bandwidth and memory usage in EO missions (Z. Zhang et al., 2019) and before the implementation of segmentation and object detection methods. Convolutional Neural Networks (CNN) have demonstrated excellent performance in various visual recognition problems such as image classification and enable accurate onboard cloud detection in small satellites.

With the increase in EO missions coupled with high-resolution modern sensors, there is an increase in bandwidth requirement that leads to the need to utilize new techniques to manage the bandwidth resources efficiently.

5.1.1. Classical approaches

In the majority of missions, all images taken are transmitted to the ground, which requires a significant amount of bandwidth. Traditionally, data collection is done by specifying in advance where and when to take the measurements. Based on the content of the data, there is no mechanism to tailor what is downlinked. (Srivastava, 2003; Vladimirova and Atek, 2002)

Other common approaches include novelty detection based on spectral contrast, radiance spatial or temporal contrast. (Shaw and Burke, 2003) But these methods are better used for dark grounds like vegetation or deserts as clouds contrast in color compared to them. Furthermore, these methods rely on manually chosen thresholds, which are time-consuming to find and sometimes brittle. (Arechiga et al., 2018)

Whereas spatial coherence is a better method of cloud detection in areas with little contrast with the clouds (ice sheets). NNs have also been shown to have greater flexibility with classifying indistinct classes like clouds on snow.

5.1.2. ML-based approaches to On-board Image Processing

For cloud detection, Zhang et al. (Z. Zhang et al., 2019) propose a lightweight DNN based on U-Net. For performance estimation of the proposed method, training and testing of the red, green, blue and infrared waveband images from Landsat-8 were used. The lightweight DNN is based on U-Net and obtained better overall accuracy while reaching the state-of-art inference speed by applying the LeGall-5/3 wavelet transform on the dataset which compresses the dataset and accelerates the network for on-board use. Zhang et al. experimental results illustrate that the proposed model maintains high accuracy after four-level compression (Z. Zhang et al., 2019). They reduce processing time from 5.408s per million pixels to 0.12s per million pixels, and average memory cost by around 30%. The suggested method takes advantage of established image compression systems in satellites to provide a good chance of onboard cloud identification based on DL, hence enhancing downlink data transmission efficiency and lowering memory costs. On compressed datasets, U-Net gives improved accuracy. In addition, the U-Net framework demonstrated tremendous promise for pixel-by-pixel categorisation of remote sensing datasets (Z. Zhang et al., 2019).

Hinz et al. (Hinz et al., 2020) also work on the detection of clouds in the H2020 EO-Alert project framework. However, the EO-Alert project aims at keeping images of clouds and enriching them with alert profiles in case of severe storms for weather broadcasting. The algorithm used is ML-based Gradient Boosted Decision Trees and is embedded in a modular image processing pipeline. Currently, tests of the pipeline are performed in Matlab and are ported on hardware to be flown to space.

Srivastava et al. (Srivastava, 2003) suggested using Kernel methods for better onboard discovery computation of cloud detection over snow and ice. This paper proposes a Kernel method that can be used for clustering and classifying images on board any satellite. The paper discusses a novel variant of the Probabilistic Kernel (P-Kernels) with a mixture of Gaussian and spherical covariance structures. It is very sensitive to even the smallest changes as it assumes all observations are independent. The results showed great promise, with clouds being differentiated much better from Greenland ice sheets compared to the Gaussian and Gaussian mixture models.

Giuffrida et al. (Giuffrida et al., 2020) (Giuffrida et al., 2022) discuss a CNN deployed on the PhiSat-1 reconfigurable nanosatellite to analyze imagery from its Hyperscout-2 payload and select images eligible for transmission to the ground. It is implemented on-board the ESA Phisat-I mission to classify cloud-covered images and clear ones. Only images with less than 70% cloudiness are transmitted to the ground. The network is trained and tested against an extracted dataset from the Sentinel-2 mission, which was appropriately pre-processed to emulate the Hyperscout-2 hyperspectral sensor. On the test set, 92% of accuracy is achieved with 1% of False Positives (FP). The results showed a power consumption of 1.8 W, requiring memory of 2.1 MB, keeping within the power and the memory constraints.

(Del Rosso et al., 2021) showcase the use of CNNs on multispectral data to detect volcanic eruptions on-board a satellite. Onboard detection of disaster events allows prioritizing their downlink and thus optimising response times, which can translate into saved lives. Moreover, they have released the dataset used for training, a step the rest of the industry should imitate if rapid progress is to be encouraged.

(Spiller et al., 2022; Thangavel et al., 2023, 2022a, 2022b) showcase the use of CNNs on hyperspectral data to detect wildfire on-board a satellite.

Other solutions that have not flown yet and are in the concept phase have been developed. Maskey et al. (Maskey and Cho, 2020) proposed an ultralight CNN algorithm called CubeSatNet, that prioritizes quality data over quantity without changing the constraints of size, power, volume, downlink and pointing requirements imposed by a 1U CubeSat. The algorithm is trained over 48000 augmented images from CubeSats and validated against 12000 augmented images from CubeSats to classify images as “bad” when cloudy, sunburnt, facing space or saturated. Images are classified as “good” in all other cases. If in orbit, the algorithm would select only “good” images to be downlinked and discard images that are covered in clouds or too bright or dark. Trained on BIRDS3 satellite images, the algorithm reportedly has an accuracy of 90% and can cut operation time by about 2/3 while significantly improving the quality of images received.

Murray (Ireland, 2019) proposed a concept of on-board processing with two cameras: the nadir-looking camera performs the standard observation, whereas a forward-looking camera observes if clouds are coming in the trajectory of the satellite. A neural net classification grid is used to identify clouds and an algorithm then decides when to capture images with the nadir looking. This approach would be oriented towards CubeSats.

Castaño et al. (Ricard Castaño et al., 2007) trained an SVM for estimating the opacity of atmospheric dust and water ice on Mars on data from the THEMIS camera mounted on board the Odyssey mission. The authors use both a regular SVM and a reduced-set SVM. The reduced-set SVM is trained on a reduced synthetic dataset maximizing the similarity of the reduced-set SVM to the regular SVM. The reduced amount of support vectors decreases compute requirements. They then test both the full-size SVM and reduced-set SVM on flight software, showing the capability of such software to run the proposed algorithms.

The authors mention two challenges related to the analysis accuracy of onboard Time History of Events and Macroscale Interactions during Substorms (THEMIS) data. Firstly, as the onboard data is not calibrated, the deployed models must be robust to significant noise. Secondly, the camera's response function can gradually increase or decrease its values due to temperature fluctuations, even when there is no change in actual value. The authors suggest characterizing the operation of the algorithms in an environment as close as possible to that of the spacecraft.

Lastly, the Autonomous and Reactive Image Chain (CIAR) project from IRT Saint Exupéry demonstrated cloud segmentation on board the operational test-bed satellite OPS-SAT in 2021 (Frédéric Férésin et al., 2021). Figure 5 showcases a visualization of their results.

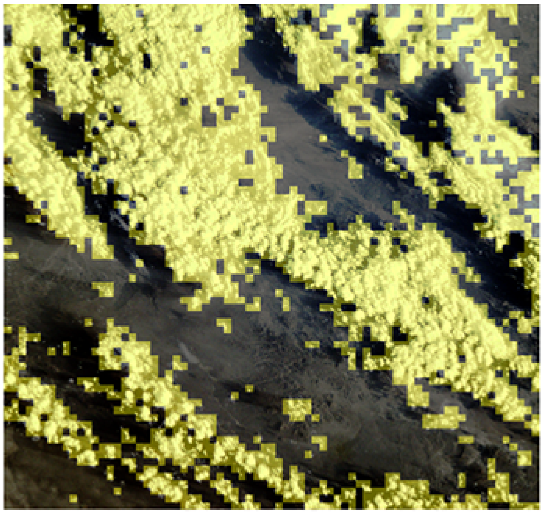

Figure 5: On-board cloud segmentation from the CIAR project

5.1.3. *ML-based approaches to Novelty Detection*

Wagstaff et al. (Wagstaff et al., 2017) show the benefits of reduced downlink data when performing cloud detection and filtering for EO missions. Cloud detection is demonstrated using Random Decision Forests (RDFs) and Bayesian Thresholding (BT), while a third saliency-based algorithm is used for novelty detection onboard EO-1. The RDF method analyzes a window of values around the pixel for classifying the pixels. In contrast, the BT independently performs the classification of each pixel. BT uses the difference in particular wavelengths between dark surface materials and bright cloudy regions. The novelty detection algorithm identifies such regions within an image that may contain new features. EO-1's primary science instrument is Hyperion. It's an Imaging Spectrometer capable of data collection with high Spatial and Spectral resolution. Using data from previous mission phases, both cloud detection algorithms were trained to drop useless images from the telemetry downstream. The performance of the algorithms has been evaluated onboard over a five-month period from November 2016 through March 2017. In comparison to ground testing, the on-board performance showed similar or better results on a diverse collection of targets. Both RDFs and BT reached an accuracy of more than 90%. However, in real-time, the RDFs were faster. The novelty detection was able to detect new features in remote locations such as small lakes and buildings; hence, such images could be given priority for the downlink. Such methods must be able to successfully operate on board with limited resources while posing a minimum risk to the overall spacecraft. With the advancement in computing capabilities, more complex models offering better accuracy can be used onboard future EO missions.

Chien et al. (Chien et al., 2017) present the results of the IPEX, which was based on a CubeSat that did fly from December 2013 to January 2015 and validated autonomous operations for the computation and generation of product onboard the platform hosting the Hyperspectral Infrared Imager (HyspIRI) mission concept's Intelligent Payload Module. IPEX was used as a testbed for on-board image classification, which was accomplished with the help of machine learning-based random decision forest algorithms. In comparison to earlier missions, the solution was improved by using an ensemble of several trees to increase the classifier's reliability through statistical regularization without the requirement for explicit tree pruning. Furthermore, the system examines spatial neighborhoods in each image rather than single pixels to integrate local morphology and texture. By classifying every 10th pixel and the vertical and horizontal directions and filling in the rest with nearest-neighbor interpolation, runtime was reduced. The IPEX classifiers are trained before launch using only four hand-labelled photos from a high-altitude balloon mission that used the same type of camera as IPEX. This is a very fascinating point. According to the researchers, it was the first time that an ML system was trained on a suborbital mission and then effectively used in orbit.

IPEX also experimented with an unsupervised method for identifying photographs with potentially intriguing content, which would be used in conjunction with supervised learning. To extract relevant regions for downlink in captured imagery, computer vision visual salience software was used. To work with CubeSat's limited resources, the program developed a simple pixel-based measurement of visual salience for grayscale images with the local context. To select the five most important parts within the image, the method is applied to a down sampled version of the image using a 32×32-pixel window. The pipeline is finished with thumbnails of important regions and their salience scores, which are saved and made available for downlink and on-the-ground analysis. If necessary, full-resolution images can also be downlinked to ground stations.

5.1.4. *Recommendations*

With the strict limitation on bandwidth, onboard filtering of useless data enables sending data to the ground with minimum compromise on image quality and the need for human intervention for decision-making. The results of ML algorithms can be improved in terms of accuracy and precision with the availability of newly generated data.

5.2. Object/Image Classification

Image classification is a task of extracting information on the basis of objects in the images instead of individual pixels, where "objects" are referred to as meaningful scene components that distinguish an image (Deepan and Sudha, 2020).

5.2.1. Classical approaches

While current methods do extensively apply ML algorithms to great success, image classification is more often done on the ground instead of onboard a satellite. (Shaw and Burke, 2003)

5.2.2. ML-Based approaches

Arechiga et al. (Arechiga et al., 2018) give an example of an on-board processing application where a CNN architecture is used for object classification and trained using satellite imagery of Planet's Open California dataset. Nvidia Jetson TX2 is used for implementing this application. The authors suggest that more research can be done so that the application can be enhanced to classify more objects. Machine intelligence is used to perform onboard analysis of EO tasks such as hazard analysis (e.g., wildfire and flood detection), target detection, area monitoring, and weather forecasting (Manning et al., 2018). On MODIS (Moderate-resolution imaging spectroradiometer) data, NASA Goddard researchers employed machine learning to detect wildfires. In practice, CNNs are used to perform two tasks: training and inference. The process of "learning" the ideal set of weights that maximizes the accuracy of the desired task is referred to as training (e.g., image classification, object detection, semantic segmentation). It's a computationally difficult task that's frequently aided by Graphics Processing Unit (GPU). The inference is the process of making decisions based on new data using a trained model (with no parameters changed). The inference is a less computationally intensive method that has been carried out on Central Processing Unit (CPU), GPUs, and Field Programmable Gate Array (FPGA).

5.2.3. Recommendations

Similar to onboard cloud detection, moving object classification and detection onboard satellite platforms allow operators to reduce the load of ground-satellite communications links. EO Operators can leverage the huge and quickly expanding research field of computer vision.

The high-level information gained by using object classification can then be used for other tasks, like dynamic mission replanning.

5.3. On-board image compression

New, complicated onboard sensors can quickly saturate communication transceiver downlink bandwidth as well as onboard data storage capacity. Image compression codecs that are more efficient are becoming a need for spacecraft and can greatly lower the amount of data communicated or stored. However, while designing a tradeoff mission, it's also important to think about whether these are computationally intensive and require quick processing to keep sensor data rates up.

5.3.1. Classical approaches

Systems used a range of lossless and lossy compression algorithms to compress data in spaceborne activities (Giuffrida et al., 2022; "Image Data Compression," 2021; "Lossless Data Compression," 2020). Where the system bandwidth is too low to support lossless compression, when the science value is not compromised by lossy compression's distortion, or when other sensors that do not play a role in primary data products are included, lossy compression is frequently used. An example of this last case can be scene-context cameras.

5.3.2. ML-Based approaches

Goodwill et al. (Goodwill et al., 2020) proposed an ML-based solution to achieve good reconstruction fidelity after lossy compression. The algorithm, CNN-Joint Photographic Experts Group (JPEG), makes use of a hybrid approach combining CNNs and JPEG Compression. The image is fed to a 3-layer CNN in the encoder to obtain a compact image representation, which is then encoded with JPEG. Based on previous work, the encoder is denoted by ComCNN and learns a compact image representation that is half the size of the original image. In the decoder, the resulting image is upsampled to the original size and decoded with a deeper 20-layer CNN, which reconstructs the original image by learning a residual image and adding it to the upsampled image.

On an image dataset obtained from STP-H5-CSP compressed to the same file size, experimental results for CNN-JPEG demonstrate a 23.5 percent and 33.5 percent gain in Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) over conventional JPEG, respectively. At a fixed PSNR, CNN-JPEG increased the average compression ratio by 1.74 times on the same dataset. It's also worth noting that the encoding segment of CNN-JPEG in TensorFlow (TF) Lite, when run on the Zynq-7020's Cortex-A9 cores, provided an average execution time of 16.75s utilizing a single thread, according to the research. Using the TF Lite interpreter to parallelize operations was reportedly far from ideal linear speedup. Authors also showed that leveraging the Zynq-7020 FPGA resources through SDSoC for hardware acceleration helped decrease the average execution time of the CNN-JPEG encoder to 2.293 s, with a 7.30 speedup over the single-threaded TF Lite solution and 6.87 times speedup over the single-threaded TF Lite solution.

Vladimirova et al. (Vladimirova and Atek, 2002) discuss the development of a lossless compression method without the drawbacks of low compression ratios using predictive NNs, coupled with integral wavelet transforms and the Peano-Hilbert (PH) Scan algorithm. This is then benchmarked against the Context-Based, Adaptive, Lossless Image Codec (CALIC) Method using various image datasets. The image is first sent through the Integral Wavelet Transform (IWT)to produce a de-correlated image, which is mapped, and a PH scan is performed after which the NN (a two-layer, $4x10^6$ x 1) scans and allocates a probability distribution for the next incoming value. On the tested data sets, using only the NN method achieved an average compression ratio of 2.530, compared to the CALIC method which achieved a ratio of 1.806. Introducing the PH scan brought an 8.5% improvement compared to the CALIC method at 2.747. The IWT+PH+NN method overall achieved an improvement of 13.1% compression ratio over the CALIC method. The paper proposes potential applications of the algorithm in previewing a satellite image before a full image is transferred to assess the image's features and would prevent bad images from being sent, such as those affected by clouds or images suffering from other distortions.

Cai et al. (Cai et al., 2003) proposed a novel Light Detection and Ranging (LIDAR) image data compression method. The method is called feature indexing where specific features are assigned to a data index system generated by DNNs. The whole program is then uploaded to onboard hardware and it stores it as a dictionary for reference. The On-Board Computer (OBC) runs a feature isolation program, identifies features, and creates a resultant dataset of pure indices based on the directory. This data set is then transmitted with the location data and then is decoded on the ground. Achieves a compression level of 99.17% and works far better than standard wavelet compression methods. The method was tested against the LIDAR data of the Space Shuttle program and achieved the above-mentioned results.

5.3.3. Recommendations:

Exploiting lossy compression to ease downlink clearly represents a path to be explored. The work by Goodwill et al. (Goodwill et al., 2020) also emphasizes the importance of advancement in the field of hardware acceleration and System on a Chip (SoC) FPGAs. Indeed, on-board inference of CNNs is computationally expensive for space platforms. Further advancements can possibly support the application of more complex algorithms even in constrained environments.

6. Machine Learning in resource-constrained Earth Observation platforms

This section addresses the topic of ML in resource-constrained spacecraft performing EO tasks. These methods represent a powerful set of enabling technologies, relevant both for the emerging interest in small satellites and to preserve the operativity of large platforms experiencing failures or operating with shared resources. Moreover, the consistent technological lag of space hardware makes considerations about reduced available SWaP almost always necessary when redeploying architectures developed for Earth-based applications into orbit.

Within the scope of this work, the constraint on resource availability will be limited to on-the-edge computational and sensing capabilities, and not extended to the data. It is also out of the scope of the section to address scheduling approaches, which optimize the availability of resources to multiple subsystems or users. This variability, however, can be also seen as a source of constraint over the available budgets.

We investigate two ways in which this adaptation to technological limitations can be implemented: optimization of the AI architecture itself, and optimization of the interplay between the model and the hardware this operates on. In general, resource-constrained platforms it is necessary to maintain a holistic view of the architecture of the software, the hardware, and the data at play.

It is worth noting that another emerging technological field presenting similar constraints to the space sector is represented by Internet-of-Things (Lane et al., 2015), where the target platforms for AI are small, low-power devices.

6.1. AI Architecture Optimization

6.1.1. Pruning

Pruning is the operation of removing or zeroing parameters of a NN model, thus reducing the network's size (Han et al., 2015). This process is generally performed by associating scores with the network's elements during training in order to select the ones to prune. The lighter model is then further trained and can be iteratively re-pruned several times. Multiple pruning strategies exist, such as varying the number and nature of items pruned, the number of iterations performed or changing the scoring criteria (Blalock et al., 2020). There are also other emerging pruning paradigms that do not rely on an iterative process (H. Wang et al., 2021) (Frankle et al., 2021).

Pruning's main trade-off is to increase computational efficiency at the cost of quality/accuracy and increased training complexity. The objective is to leverage compression rates of 4, 8 or even 32 while costing at worst only a few percent of accuracy (Blalock et al., 2020). Performing pruning along this objective remains a delicate task as literature demonstrates that keeping good performances is dependent on the pruning method. The main challenge of implementing pruning is thus to determine and test which pruning methods to use in order to achieve the required compression while keeping acceptable performances for a representative type of datasets.

Although the lack of standards in evaluation impedes the comparison of the multiple existing studies, they all advertise significant compressing at low accuracy cost, including several algorithms confirmed by multiple papers (Blalock et al., 2020). Pruning has been successfully applied in many image processing use cases but has also been proven on voice processing (He et al., 2014), credit classification (Tang et al., 2018), and multiple other types of datasets (Lazarevic and Obradovic, 2001).

Additional engineering and more complex training on the ground in order to significantly reduce the onboard execution constraints make pruning an attractive trade-off and a strong technological enabler of NN implementation in space.

Pruning is now developed enough to have documented implementation and examples in ML frameworks such as TF ("Pruning in Keras example | TensorFlow Model Optimization," 2022).

So far, pruning has been used as part of complex NN applications for space but only on the ground with applications such as image classification (Browne et al., 2020; Castelluccio et al., 2015; Kavzoglu and Mather, 1999; Maggiori et al., 2017). There are some applications aiming towards on-board implementations like remote sensing image classification (Pitsis et al., 2019; Zhang et al., 2020), vehicle detection in satellite images (Tan et al., 2020) and image anomaly detection (Ma et al., 2019).

Unfortunately, the authors were unable to find documented evidence of a pruned NN that flew on a space mission.

6.1.2. Filter compression and matrix factorization

In its section concerning "convolutional filter compression and matrix factorization," the paper by Goel et al. (Goel et al., 2020) presents methods to adapt neural networks to low-power platforms by operating at a layer's level. The distinction operated between the two distinguishes between the types of network elements that are being optimized.

Neural Networks can be algebraically represented as n-dimensional matrices known as tensors. Matrix factorization approaches reduce the complexity of these underlying tensorial structures, to obtain compressed networks without significant loss of accuracy. Filter compression methods, on the other hand, reduce the number of parameters in the network architecture by acting on the structure of filters in the so-called convolutional layers.

In particular, Goel et al., observe that filter compression methods are capable of achieving state-of-the-art accuracy in computer vision, albeit at times at a high computational cost. As computer vision tasks are essential in EO operations, this class of methods appears to be the most significant within the scope of this paper.

Two architectures emerging as relevant for filter compression are SqueezeNet (Iandola et al., 2016) and MobileNets (Howard et al., 2017). Both these architectures have found applications in the EO community. For example, modified SqueezeNets have been used by Haikel (Haikel, 2018), Alswayed et al. (Asmaa et al., 2020) and Alhichri et al. (Alhichri et al., 2018)for the classification of remote sensing images (both in drone and satellite images). In particular, Alswayed et al. report results comparable to or outperforming the state of the art at the time of publication.

Poortinga et al. have used a MobileNet-based architecture to map sugarcanes in satellite data of Thailand (Poortinga et al., 2021), obtaining significant accuracy for the task. Zhang et al. (B. Zhang et al., 2019) also have used an architecture capitalizing on MobileNet, reporting results outperforming the state of the art at the time. Similarly, Yu et al. (Yu et al., 2020) present a MobileNet-based method to classify remote sensing imagery and report outperforming many state-of-the-art models while requiring a smaller amount of training data. In their report paper, Hoeser et al. (Hoeser et al., 2020) note that: “It is important to note the small group of six items which use MobileNets, of which five were published in 2019”. They describe an onset of interest in parameter efficient models with high accuracy and they prove that such models can compete in Earth observation studies.

6.1.3. Architecture search

Neural Architecture Search (NAS) refers to a set of tools and processes for the automatic generation of optimal architectures for an ANN. NAS is a specific instance of automated machine learning (AutoML), the process of automating the overall ML construction process (He et al., 2021). As shown by Chan et. al (Chan et al., 2018), this process can be specialized to address a constraint on available resources.

Seminal developments in NAS emerged in late 2016, from the work of Zoph and Le (Zoph and Le, 2017) and Baker et al. (Baker et al., 2017). In a survey on the subject, Elsken et al. (Elsken et al., 2019) report three key parameters to operate a classification of NAS processes. These are:

- Search space,
- Search strategy,
- Performance estimation strategy.

Being an approach to adapt the heavy computational cost of NN to resource-constrained platforms, NAS has naturally found application in many space-related use cases. EO, there has been quite a research on hyperspectral images classification using NAS, with development performed by Liang et al. (Liang et al., 2020) have employed NAS (and pruning) to detect aircraft in remote sensing images. Mobile Neural Architecture Search (MNAS) (Tan et al., 2019) is a probable candidate in implementing NAS to EO satellite inference on the edge application.

6.1.4. Knowledge Transfer and distillation

In Knowledge Transfer (KT) and Knowledge Distillation (KD) a small, lightweight network is trained to reproduce the behavior of a large, computationally intensive network without having to duplicate the architecture of the latter fully. This leads to small networks both providing results comparable to those of large networks and deployable on resource-constrained platforms.

According to the paper of Goel et al. (Goel et al., 2020), in KT the smaller network is trained using data labeled by the larger network (defined as “synthetically labeled data” by Ba and Caruana (Ba and Caruana, 2013)), while in KD a small network (student) is trained by a large network (teacher) to replicate the latter’s output. Within the scope of this section, it also appears relevant to discuss transfer learning, which has attracted considerable interest from the space community.

De Vieilleville et al. (de Vieilleville et al., 2020) proposed a distillation method to perform DNN-mediated segmentation of EO images on board of CubeSats. In this work, they show that a 10 to 30-fold reduction of the free parameters of the network mediated through distillation leads to weakly worse performance (+5/-10% accuracy). Similarly, (Chen et al., 2018) provide a detailed distillation implementation and results showing a strong reduction of the NN execution load while keeping a steady accuracy in remote sensing scene classification. (Bazzi et al., 2020) applied distillation for mapping irrigated areas using remote sensing data.

Since 2019, self-distillating networks are emerging (Chen et al., 2021) with one successful implementation for cloud detection in remote sensing by (Chai et al., 2020) achieving 200-fold compression. Industrialization is not as developed as pruning as there are only a few open access examples of implementations but no widely developed library. Unfortunately, the authors were unable to find documented evidence of a distilled NN ever flown and used on a space mission.

6.2. Hardware Acceleration

Computing limitations are demanding to ML-based applications because of the significant amount of data to be processed for DL. Many NN models require high-end GPU devices to run in inference, and even more so during training. In deploying ML to an EO satellite, it is appropriate to consider the inferencing phase due to volume, power, and mass constraints, especially under CubeSat standards. Progress in commercially available off-the-shelf hardware in mobile edge computing has a progressive effect in finding their way to CubeSats in implementing DL algorithms for space applications (Kothari et al., 2020).

With CPUs considered to be general-purpose computers, AI-specific hardware such as GPU’s, FPGA, and Application-Specific Integrated Circuit (ASIC) takes the center stage which is designed to accelerate the computation of linear algebra and specializes in performing fast and matrix multiplications with higher performance-per-watt ratios. Furthermore, advanced next-generation architecture for onboard computing which heavily depends on artificial intelligence is developed like Artificial Intelligence-Onboard Computing (AI-OBC) (Huq et al., 2018) based on distributed on-board architecture consisting of CPU, Visual Processing Unit (VPU), emerging AI accelerator class of microprocessor for running machine-learning applications to train DNN and FPGA connected through CubeSat Service Protocol (CSP) through which ML and training are carried out in real-time with COTS components to reduce cost and development time. One other form of tailored hardware optimization is the adoption of spiking neural networks (Kucik and Meoni, 2021) and their deployment on optimized hardware. This approach, which is much closer to the way the brain seems to function, can allow for dramatic energy savings through minimization of energy use during neuron activation.

Table 1: Hardware Accelerators

Name	Company	Description
Intel Movidius Myriad 2 Vision Processing Unit (VPU)	Intel	Implemented with DNN in Phisat-1 (Esposito et al., 2019)
Myriad X (VPU)	Intel	Active testing (Bruhn et al., 2020)
Jetson Nano (GPU)	Nvidia	Space Edge Zero (2021) by Spiral blue (Mittal, 2019)
Tegra TX1 and TX2 (SoC)	Nvidia	Demonstrated AI Image processing capability (Buonaiuto et al., 2017; Hernández-Gómez et al., 2019)
Coral TPU	Google	Used with SC-LEARN Architecture for Hyperspectral models (Goodwill et al., 2021)
Apache 5	Almotive	In development
Neuromorphic chip	Innatera	In development
Spaceborne Computer-2 (SBC-2)	Based on Intel Xeon	Onboard ISS
Ultrascale Radiation Tolerant (RT) Kintex FPGA	Xilinx	Prototype available
Xilinx Zynq-7020 (ARM Cortex-A9 + FPGA)	Xilinx	Space Test Program Houston 5/ CSP (2017)

6.3 Quantization / BNNs

In quantized networks, the number of bits used to represent numbers defining a model is reduced. This provides a decrease of orders of magnitude in computing, memory and power requirements, for a comparatively low decrease in performance. Quantization may be applied to weights, activation functions or gradients of a network, either during or after training. (Guo, 2018; Qin et al., 2020; Simons and Lee, 2019). Quantization has been explored in research for remote sensing image segmentation and processing but appears to never have been flown in space.

Perhaps the most common established quantization technique is reducing the bit-width of weights after training. However, very low bit widths, typically of four or less, usually incur heavy losses. This can be mitigated by performing model training under the reduced bit-width quantization, known as Quantification-Aware Training (QAT). Good results have been achieved with quantization, even going all the way to a single bit.

Accuracy on par with full-precision NNs was achieved for standard datasets in publications such as Binary-Connect, Exclusive-NOR Network (XNOR-Net), and Trained Ternary Quantization (TTQ) (Courbariaux et al., 2015; Rastegari et al., 2016; Zhu et al., 2017). Quantization of already existing NNs such as AlexNet (Krizhevsky et al., 2012) and Visual Geometry Group Network (VGGNet) (Simonyan and Zisserman, 2015) applied to the ImageNet dataset has been carried out without any accuracy loss while reducing their sizes up to 50 times (Han et al., 2016). Quantization both before and after model training is provided today either as part of mainstream DL libraries ("Post-training quantization | TensorFlow Lite," 2022.; "Quantization — PyTorch 1.9.1 documentation," 2022.) or third-party libraries such as Larq ("Larq | Binarized Neural Network development," 2022.) and FINN (Alam et al., 2022) respectively.

Although there exists no consensus on why quantization works, a candidate explanation argues that large amounts of pathway redundancy in NNs make the expressivity loss a minor concern. Theoretical analysis in that regard is still limited. Anderson and Berg (Anderson and Berg, 2017) found that statistical properties of the computation are kept even when a network is binarized. Molchanov et al. (Molchanov et al., 2017) indicate that nearly 99% of weights can be pruned in certain NNs and achieved a 68-times sized reduction on VGG-like networks without loss of accuracy.

Quantization techniques can be divided into two main categories: Deterministic and Stochastic. Guo classifies deterministic quantization methods (Guo, 2018) into:

- **Rounding:** Floating-point values are assigned their nearest fixed-point representation.
- **Vector Quantization:** Weights are clustered into groups, with the centroid of each group replacing the real weights.
- **Quantization as an optimization:** Here, the quantization is treated as an optimization problem, which involves minimizing an error function taking into account real and quantized weight values.

Regarding stochastic quantization techniques, they separate them into:

- **Random Rounding:** The quantized value is obtained by sampling a discrete distribution parameterized by the real values themselves.
- **Probabilistic Quantization:** Weights are assumed to be discretely distributed, with the methods trying to estimate which distribution function it is.

Deterministic quantization has seen extensive success, with rounding being the most commonly successfully employed type of quantization, such as Rastegari et al. (Rastegari et al., 2016) and (Polino et al., 2018), where a general rounding function was introduced. In particular, Binary-Connect Courbariaux et al. (Courbariaux et al., 2015) used binary rounding, achieving 98.8% accuracy on the MNIST dataset. Also noteworthy is the use of vector quantization in Gong et al. (Gong et al., 2014), where a network compression ratio of 24 was obtained, losing only 1% of accuracy on the ImageNet dataset. However, Stochastic quantization has not experienced such a resounding success, perhaps due to an over-reliance on statistical assumptions which are not guaranteed to hold.

Quantization approaches may quantify several or all of the following:

- **Weights**: The action of quantizing weights yields a smaller network size and can accelerate the training and inference process. However, this comes at a price: NNs will have a harder time converging when training with quantized weights, and a smaller learning rate is required. Additionally, the gradient cannot back-propagate through discrete neurons, leading to the use of straight-through estimators in order to estimate them, usually with a high variance.
- **Activations**: The goal of quantized activations is replacing inner products with binary operations, reducing memory constraints since the operation precision is reduced, all while accelerating network training. In fact, activations may fill more memory than weights (Mishra et al., 2017). Note that quantized activation will cause what is called a “gradient mismatch”, where the gradient of the activation function is different from the one obtained from the straight-through estimator used.
- **Gradients**: Quantizing the gradients is still a relatively new avenue of research in NN quantization. The main objective here is not reducing the model size, but aiding in distributed network training, where several computing nodes need to share information of the gradient values between them. The smaller the size of the data the nodes need to share, the faster parallel training can be performed. Quantized gradients need to be carried out with care since unsuitable implementations run the risk of causing the gradient descent algorithm not to converge.

7. Machine Learning standardization and issues in Earth Observation Operations

Interest in AI and ML has increased in the past years. Many groups in different industries are working on creating guidelines, best practises, and standards to help make sure these systems are used correctly. But the process is far from over, and so far, the space industry has only given us a real-world example of something similar. Standards, guidelines, and other documents discussed in this section blur the line between definitions of AI and ML. While we find this fact misleading, we have kept the original usage from the sources in order not to alter their message.

These bodies of work aim at aiding ML system developers to avoid common pitfalls and problems associated with these systems. We provide in this section a cursory overview of what these problems are in order to raise awareness amongst EO platform operators. We do this so that the designers and operators of EO platforms are aware of what ML systems are capable of and are not capable of doing when it comes to making decisions that are reliable, intelligible, and appropriate for usage in situations with high stakes. Because of the high expense of these possible applications on-board a big satellite platform, these conditions apply to the majority of those applications. To put it another way, any operator who is contemplating delegating decisions regarding the success or failure of their mission to ML systems should make it a priority to employ ML systems that are reliable and can be explainable.

We do this so that EO platform designers and operators are aware of what ML systems can and cannot do when it comes to taking decisions that are trustworthy, understandable, and fit for use in high stakes scenarios. These conditions apply to many of the potential applications on-board a large satellite platform, due to their cost. In other words, using trustworthy, explainable ML systems should be important to any operator thinking of charging such systems with decisions deciding the success or failure of their mission.

7.1. Guidelines and Roadmaps

International Standards by International Standardization Organization (ISO) committee ("ISO - ISO/IEC JTC 1/SC 42 - Artificial intelligence," 2017.) are currently available or under development. These standards and projects represent the united efforts of experts and entities in providing guidance and focus on the standardization of Artificial Intelligence, with currently more than twenty under development and six already published. We found ISO/IEC TR 24030:2021 to be particularly interesting as it covers 132 use cases, as well as the projects under development concerning Functional Safety and AI, data quality and AI explainability. The ISO is not alone in working on AI standardization, though.

The Chinese Big Data Security Standards Special Working Group of the National Information Security Standardization Technical Committee (NISSTC) wrote the Artificial Intelligence Security Standardization White Paper (Törnblom and Nadjm-Tehrani, 2019). The focus of this White Paper ranges from the security of AI to main security threats, risks, and challenges. Seven recommendations have been made on the importance of improving a system of AI security standards, the need to speed up the development of standards in key areas, promoting the application of AI security standards, strengthening the training of AI security standardization talent, participating in international AI security standardization, establishing an AI high-security risk early warning mechanism, and improving AI security supervision support capabilities.

Germany developed an Artificial Intelligence Standardization Roadmap (Wahlster and Cristoph Winterhalter, 2020), continuously updated, as a joint effort between DIN and DKE. The roadmap strongly supports the idea that standardization would improve the explainability and reliability of AI, thus favoring its application. In the roadmap, AI's explainability and reliability, they deal with data reference models for the interoperability of AI systems, development of an AI basic security standard, practice-oriented initial criticality checking of AI systems. In addition, the work provides extensive analysis on the definition of AI as well as classification schemes to evaluate AI-based systems.

The work is particularly interesting also for spotlighting issues as the risk-based assessment of applications, trustworthiness, ethical approach and AI application lifecycle. In addition, in each section of the roadmap, specific needs in the direction of standardization are pinpointed.

The European Commission (EC) shaped a white paper ("White Paper on Artificial Intelligence," 2020.) setting out policies to achieve the uptake of AI in the European Union (EU) and to address risks associated with the use of AI technology. Along the sections of the document, it gives particular attention to the opportunity to create an ecosystem of excellence. Six actions have been highlighted, among which: focusing on SMEs and ensuring that each member state has a digital hub highly specialized in AI; strengthening public-private partnerships in AI, data and robotics; and promoting the use of AI in the public sector. An overview of the most significant risks is also provided, with more emphasis on ethical and trustworthy AI.

The National Science and Technology Council from the USA's Executive Office developed an AI Research Development Plan in 2016, later updated in 2019 (Faisal D'Souza, 2019). The Plan does not define specific research agendas for Federal agency investments but highlights strategies to reach a given portfolio. While it must be noted that the utmost focus of the strategies is not on the standardization, strategy 4 "Ensure the Safety and Security of AI Systems" and Strategy 6 "Measure and Evaluate AI Technologies through Standards and Benchmarks" are covering aspects strictly related to standards and certifiability. It is worth mentioning great attention to the development of shared public datasets and open-source libraries, as means to keep the technological lead.

Although slightly different in scope, as more oriented towards certification rather than standardization, it is worth mentioning the White Paper (Gregory Flandinet.al., 2021). The document aims at "sharing knowledge, identifying challenges for the certification of systems using ML, and fostering the research effort". A thorough discussion on the features that an ML-based system should possess to be certified is carried on, leading to the identification of seven challenges to tackle: probabilistic assessment, resilience, specificality, data quality, explainability, robustness, and verifiability.

7.2. Issues and Techniques

In this section, we offer a brief discussion of the potential unique issues one may encounter when developing and operating a system that incorporates ML. Whenever possible, we discuss some current approaches to bridge these issues. This discussion is meant to be illustrative to the reader and an encouragement to explore the topics in further detail, but it attempts to be comprehensive on neither scope nor depth. Furthermore, the topic is under active research and is likely to expand in the coming years.

7.2.1. Explainability

ML models, and particularly large models with lots of free parameters such as large decision trees or NNs, can act as black boxes. The process by which they arrive at the final output can be too complex to be directly interpreted, thus becoming as inscrutable as if the model's internals had been inaccessible in the first place.

However, transparency, explainability, and interpretability are very important for any technical system with a moderate or large impact, be it in terms of dollars or human lives. Therefore, model explainability is very important in fields such as aerospace, medicine, insurance, banking, and more.

Explainability is a hard problem because of several reasons. Firstly, it is user-dependent: the type of explanation expected by an average user will differ from that expected by a regulator or an engineer. This leads to the question "How detailed must the explanation be, and what must it cover?". Secondly, the expected outcome of transmitting an explanation can be hard to define, should the receiver become more able to predict model output after receiving explanations? Must the explanation point univocally to the features of the input data that had the largest impact on the produced results, and is this limited to input data, or does it also include training data? Perhaps it should illustrate a counterfactual - «What would need to change for the decision to have been different? » Or perhaps something else entirely? And are the previous goals mutually exclusive?"

There are a huge number of techniques to answer some of these and related questions. The field of Explainable AI (XAI) for short, is huge and expanding rapidly. Providing an overview of this field is not within the scope of the current publication, but we recommend our readers to consult the Interpretable Machine Learning book (Molnar, 2021) or one of the numerous reviews on the topic to learn more (Linardatos et al., 2021; Tjoa and Guan, 2020).

7.2.2. Robustness and reliability

Reliability is the rate of failure of a system when operating in nominal conditions (e.g. 10^{-9} catastrophic failures per flight hour ("AC 25.1309-1A - System Design and Analysis – Document Information," 1988)). Since a rate of system error can be extremely challenging to calculate without operating the system, heuristic development rules like no single point of failure are accepted as valid ways to achieve the goal. This acceptance stems from either a competent authority, which implicitly accepts the risk of not properly achieving the desired reliability level or historical data when available. Neither is a possibility for current ML-based systems, due to an absence of historically validated, robust, and widely accepted heuristic design rules.

For Machine Learning systems, reliability comes from two distinct factors: accuracy and robustness. An ML classifier with higher accuracy is less likely to misclassify an input, hence is more reliable. Performance does not usually come into play for classical software system's reliability as accuracy for a valid set of inputs and execution path is 100%. This section does not concern itself with increasing model accuracy, a topic that is the main focus of each application-specific research field mentioned so far.

Accuracy for an ML model is calculated over the data points in the test dataset and only those. While this is also true for classical software testing, in the latter the notion of input equivalence classes provides assurance that the software system will continue to perform acceptably for inputs outside the test set. Correctness equivalence classes for ML models do not currently exist. A similar notion of robustness can be used instead. A robust model has bounded accuracy loss for inputs that are within a bounded distance of the input distribution. This fact can be used to construct arguments for the reliability of an ML model.

Equivalence class discovery for random forest models is a topic under active research (Cheng and Yan, 2021; Törnblom and Nadjm-Tehrani, 2019).

When demonstrating model robustness, several problems arise:

Firstly, how does one quantify the distance between input data? Although several measures exist, they are often hard to relate to humans' tacit notions of input distance. It is easier to qualitatively say to what degree an image does not depict a cat than it is to quantify it in a single measure. This only becomes harder for more abstract forms of input such as satellite telemetry data. Thus, relating system-level specifications to notions of input distance is sometimes complex. For a given distance definition, formal verification methods attempt to formally prove certain properties of DL models, including robustness (Katz et al., 2017; Mirman et al., 2018; Müller et al., 2021; Wang et al., 2018). They allow a user to build a model tolerant to a certain distance between inputs. Equivalent research exists for other ML models, such as random forests (RFs) (Törnblom and Nadjm-Tehrani, 2019), but the literature is significantly less developed. Note that these approaches allow a designer to fight adversarial examples, a specific and concerning failure mode for ML models (Chen et al., 2019; Goodfellow et al., 2015). Nonetheless, the literature on the generation and defeat of adversarial examples is highly active and ever-evolving, as measures, countermeasures, and counter-counter-measures get deployed. It is out of scope for this review to delve any deeper into that.

Secondly, there is the well-known issue of generalization. A model may offer very good performance on a dataset and very poor performance on the actual population, in the phenomenon known as overfitting. The PAC-Bayes approach offers generalization bounds that specify a minimum number of samples from distribution for a desired performance and training process reliability levels within that distribution. These bounds, however, are often extremely conservative, and improving them is another active field of research (Shalev-Shwartz, 2014). Since it is hard to quantify these bounds appropriately, the only recourse for organizations to ensure performance is to collect massive amounts of data, which is prohibitively expensive or downright impossible in many cases. Since generalization and robustness shortcomings are highly model-specific, one approach to tackle them focuses on applying mixtures of models working in tandem, known as ensemble models, and selecting an output based on the collective response of the ensemble (Pang et al., 2019; Yang et al., 2021).

Thirdly, and also related to the second issue, there is the phenomenon of domain drift (Shweta, 2019). Models do not just overfit to a given dataset but also to the current population. And, as time goes by, systems change. An FDIR system monitoring battery health will see its voltage decrease over time as the battery ages. The statistical distribution of deviations around the nominal value is also likely to change. The performance of the ML model will thus decrease over time as the world changes around it. Fine-tuning on new data can mitigate this issue but can trigger the phenomenon known as catastrophic forgetting (Nguyen et al., 2019), where the model loses performance on old and new data. A solution is to retrain it from scratch on new data, but this entails capturing that data and retraining the model, which increases operating costs and risks in hard to predict ways. Alternative solutions exist but they come with their own drawbacks. Training a model on a dataset representative of the whole system's life cycle can mitigate the issue but requires larger models and better data capture at the project's start.

Lastly, models also overfit the specifics of the system they're trained for. A model trained for one specific satellite may have issues adapting to another satellite instance, or model. Version improvements such as equipment changes may bring performance hits with them too. While research fields like transfer learning, domain adaptation and domain generalization (Zhao et al., 2020) attempt to address the issue, they are far from universally reliable at the moment. This is particularly concerning for the space industry, where mass manufacturing and standardized equipment is the exception rather than the norm and can pose a serious challenge to the industry's adoption of ML technologies. Sometimes, when adapting to new platforms, new input data will be available or new output data may be required. In this case, the field of transfer learning is applicable, which includes both domain adaptation and domain generalization.

In short, despite the aforementioned techniques, ML models are extremely brittle to deviations in input data from the training dataset, and it can be assumed that deviations from the training dataset will break the system. Therefore, building proper datasets is a key task of any ML system designer or operator, a topic which we address in the next section.

7.2.3. Dataset Construction

Datasets are the lifeblood of ML. Therefore, it is only right to have standards assigned for data to avoid anomalies and have a perfect collection that will help produce the right results.

Cappi et al. (Cappi et al., 2021) propose a Dataset Definition Standard (DDS), which, while not specifically geared toward space activities, can be applied to EO data from either payload or satellites. It aims to provide a standard for training, validating, or testing datasets. It explains in detail the recommendations to be followed while collecting data and how to annotate it and perform functions. The paper talks about many important aspects any dataset should possess, from how it must cover as many situations as possible that could be encountered during model development to how a history of every single change to every data must be kept helping with traceability and avoid discrepancy. The paper provides clear recommendations for labeling and annotation of data and how the dataset should be segregated for training, validation and testing.

The US Geological Survey (Larry R et al., 2019) provides dataset standards for their various operations like Biological, Climate and Forecast and Mapping. Cleansing "dirty data" is mentioned as a common problem faced by data scientists. They also take it a step further with geological mapping by producing a set of parameter standards to be followed while collecting data which define a set of rules for individual parameters within the dataset. The parameter standards cover a wide range of qualities like the date/time, geographic coordinates, codes, etc. the satellite data should contain. Report (Larry R et al., 2019) explains how exactly a topographical map of anything in the US should be produced and one important aspect of it is the data standards including which standards the file formats of the data should be stored in.

For the data quality standards, they delve into it by discussing various components like currency, consistency, completeness, and accuracy. The paper covers every aspect of mapping data from dealing with off grid and oversized maps, data sources and resolutions to how cartographic features should be interpreted.

The primary space operation in Earth Orbit is remote sensing. As a result, they are the primary data-producing activities. Therefore, remote sensing standards are relatively well developed when compared to other ML operations in the space industry. Authors (Di, 2008; Liping Di and Ben Kobler, 2000) go in-depth about all standards of remote sensing including the dataset standards.

Di and Kobler (Liping Di and Ben Kobler, 2000) introduce NASA's well developed EO Systems' Data Information Systems (EOSDIS). As the EOSDIS will process data from various fields it is not feasible for the system to deal with every single data collected one by one. This has led to EOSDIS establishing standards to deal specifically with remote sensing data.

7.3. Recommendations

As outlined above, ML systems face a number of issues precluding their application in many scenarios where they would otherwise be useful.

We believe the fundamental research being carried out on ML model robustness is of great interest and recommend that any practitioner follow it closely. For certain small-scale problems, work on formal verification of ML models may already be enough to ascertain that the network responds appropriately within the input regime, and input data outside of this regime can be purged by data verification systems implemented in classical software. Further, we recommend that any practitioner keep a careful watch for ways in which the lifecycle operation of a system will deviate from the training scenarios, and mitigate the risks issued from model brittleness to these differences. The system must undergo a verification process to be verified and validated. The critical levels of various ML models are displayed in Table 2.

202 **Table 2:** ML Certification Criticality levels (Winter et al., 2021).

Criticality Level (CL)	**Impact Potential (Examples)**	**ML Application Requirements**
1	There is no risk of harm to living beings, no risk of loss of confidential data, and no ethical or privacy concerns.	Basic minimum requirements of a competently developed ML application are fulfilled.
2	Living beings could be harmed with limited, no permanent damage. Temporarily unavailability of non-critical data and services, violation of ethical concerns without identifiable harm to actual persons.	The ML application is developed according to industry standards and follows best practices that are regarded as state of the art.
3	Living beings could die or be restricted for life; the environment could be damaged. Manipulation of data with severe financial consequences and loss of control of the system to malicious attackers.	The ML application is developed and documented with great care. Safety & Security is ensured with processes and techniques that go beyond traditional best practices and industry standards.
4	Many living beings could die or could be restricted for life; the environment could be damaged permanently. Loss of information which endangers the existence of the organization. Long-term unavailability of critical data or services without which the organization cannot function.	The ML application is developed and documented with great care. Safety & Security is ensured with processes and techniques that go beyond traditional best practices and industry standards. All components of the ML application are formally secured and validated.

Major certification topics must be re-examined, even though known certification procedures for traditional applications cannot be used in a clear manner in the context of ML. This technology's total effectiveness and safety would be enhanced with a comprehensive certification strategy for machine learning applications, which would boost public acceptance and trust. Winter et al (Winter et al., 2021) proposal for certification criteria for supervised learning with low-risk potential is shown in Figure 6.

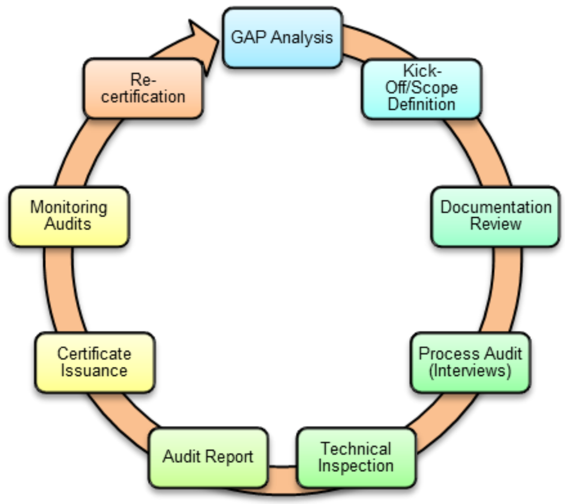

Figure 6: ML Certification workflow (Winter et al., 2021).

ML explainability is another core issue; explainability of model decisions can and does take precedence over model performance in scenarios with high-impact decisions or where (human) learning from the model's decisions is key. Current model explainability methods can offer insight into the relevant features of input data used for a model's decision, but they can also provide misleading or unhelpful signals. For applications where explainability is an important feature of the system, dictionary, tree, or kernel-based models and other easily explainable methods should be compared with harder-to-explain models for a performance-explainability trade-off.

National recommendations, white papers and initial official standards in the AI and ML field attest to the growing interest in the subject. While the scope of these is much broader than the space sector alone, some considerations can be applied to ML for space applications too.

Data quality and availability will play an important role in the adoption of ML across EO Operations and will certainly be demanded by supervisory and regulatory agencies performing standardization and certification. This need goes beyond the mere abundance of data. Relevance, cleanliness, and useability will require careful attention and control. The industry can leverage work from other fields such as the aforementioned dataset standards to achieve this. Publicly available datasets can also be a boon to adoption, such as those listed in (Cole, 2022; Rieke, 2022).

8. Discussion & Conclusion

In the area of ML in EO Operations, this evaluation effort covers different aspects, including ground operations, enhanced GNC, on-board image processing, FDIR, and standardization. It examined the state of the field, which serves as a baseline, and brought to light intriguing trends.

We have discovered that there is mounting evidence in numerous application sectors that EO missions can benefit from ML usage on-board. Case studies uncovered have demonstrated advancements in platform autonomy and performance. New capabilities, such as automated payload data filtering by sending only pertinent photos to the ground, can lower downlink bandwidth requirements, which is crucial for smaller satellites but also lessens radio frequency band saturation. Better visual-based processing also makes it possible for spacecraft to navigate using their visual systems, and RL shows promise in developing more effective nonlinear controllers. Better autonomous decision-making for EO missions is made possible by autonomous FDIR operations, allowing current teams to manage more operations more efficiently and lowering satellite operating costs.

On-board processors must meet high criteria imposed by ML algorithms. A significant difficulty is the need to optimize ML models for space applications at the hardware and software levels. The good news for space platform operators is that this reflects and exemplifies the considerably more difficult task of installing ML on edge platforms. The community can benefit from a sizable and growing body of knowledge and expertise.

From the perspective of on-board EO applications, ML has mostly been used for cloud detection and novelty/change detection. These applications frequently use vision-based techniques. EO applications could learn technical knowledge from other technical disciplines that have extensively researched vision-based ML methods and solutions. There are a few examples of SAR-based images as well, though. This would imply that there is still an opportunity for advancement and growth of these sensors in all-weather, all-day usage.

A parallel but closely related track to research and applications is being standardized. There are currently no established standards for ML in the space industry. Key areas, including explainability, robustness, and data structure creation, are the subject of rigorous research. EO Operators creating ML applications ought to make use of this area for improved performance and dependability. These research areas should be taken into account by organizations intending to publish standards and guidelines, but they must be avoided at all costs to prevent over-prescription of remedies that might compromise the success of standardization development.

Another important element that unites all ML-based EO Operations is the availability of data. The ability to use more data for machine learning in EO operations might significantly advance technology and benefit all participants, including business, academia, and space agencies. There are not many open datasets available right now, and those are mostly designed for image processing or visual navigation applications. The technological improvement favored by open datasets in a wider range of applications is a significant long-term goal for the space sector, even while it may be counter to a particular organization's short-term goals to disclose private data. We think the field should concentrate on producing and disseminating such open datasets, and we encourage players without a profit motive like space agencies, to take the lead in attaining this goal.

The fact that currently, few EO missions have used ML in orbit is a common finding across the subtopics of this review. This can be ascribed to the space industry's lengthy lead periods and slow cycles, which contrast with other sectors, such the automotive industry, which have embraced the technology. We anticipate that these cycles can be sped up as technology demonstrators move quickly from test benches to orbit with the advent of New Space and faster access to space. Further enhancing the efficiency of ML deployments in the EO Operations industry and the space industry at large, increasing the number of missions can result in better data collecting and platform standardization.

Based on the preceding critical review, it is quite evident that incorporating ML into EO operation can maximize its potential and promote additional study. The following key topics will be the focus of EO research in light of current trends and requirements:

1. Investigating the Machine Learning-based Mission Planning and Scheduling (MPS),
2. Examine the potential for Machine Learning techniques to improve Guidance, Navigation and Control (GNC) in space operations,
3. Examine the potential of ML techniques to assist with on-board data processing (OBDP),
4. Explore the effectiveness of incorporating ML models into resource-constrained platforms,
5. Investigate the effectiveness of Fault Detection, Isolation and Recovery (FDIR) using Machine Learning techniques.

Not to mention, we have found that the use of ML for EO operations frequently lags behind the state-of-the-art. Transformer models on sequential and other data are one example of a technique that has achieved significant success in research and operational environments but has not yet been publicly used for EO Operations issues. Similar to formal verification and other verified robustness techniques, there are very few applications for resource reduction strategies like pruning, distillation, or quantization.

Researchers and operators can use this critical assessment as a resource for further ML deployment and experimentation in demanding, complicated future EO missions that are more autonomous, communicate only useful data, and require much less involvement.

Acknowledgements

The authors would like to acknowledge the three partner organisations: Mindseed, Satsure, and CNES.

Mindseed ("Mindseed," 2022.) is a company focused on bridging the gap between space and non-space organisations and showcasing the benefits of space for all.

Satsure ("SatSure," 2022.) is an innovative decision analytics company leveraging advances in satellite remote sensing and Machine Learning to achieve the United Nations Sustainable Development Goals.

CNES ("cnes | Le site du Centre national d'études spatiales," 2022.) is the French National Space Agency, with activities all over the space value chain. Their Earth Observation Operations teams provided invaluable feedback for our research.

Our three partners reviewed our research, and we are deeply thankful for their collaboration.

References

AC 25.1309-1A - System Design and Analysis – Document Information, 1988. URL https://www.faa.gov/regulations_policies/advisory_circulars/index.cfm/go/document.information/documentid/22680 (accessed 9.11.21).

Alam, S.A., Gregg, D., Gambardella, G., Preusser, M., Blott, M., 2022. On the RTL Implementation of FINN Matrix Vector Compute Unit.

Alhichri, H., Alajlan, N., Bazi, Y., Rabczuk, T., 2018. Multi-Scale Convolutional Neural Network for Remote Sensing Scene Classification, in: 2018 IEEE International Conference on Electro/Information Technology (EIT). pp. 1–5. https://doi.org/10.1109/EIT.2018.8500107

Anderson, A.G., Berg, C.P., 2017. The High-Dimensional Geometry of Binary Neural Networks. ArXiv170507199 Cs.

Arechiga, A.P., Michaels, A.J., Black, J.T., 2018. Onboard Image Processing for Small Satellites, in: NAECON 2018 - IEEE National

Aerospace and Electronics Conference. pp. 234–240. https://doi.org/10.1109/NAECON.2018.8556744
Arechiga, A.P., Michaels, A.J., Black, J.T., 2018. Onboard Image Processing for Small Satellites, in: NAECON 2018 - IEEE National Aerospace and Electronics Conference. Presented at the NAECON 2018 - IEEE National Aerospace and Electronics Conference, pp. 234–240. https://doi.org/10.1109/NAECON.2018.8556744
Asmaa, A., Haikel, A., Yakoub, B., 2020. SqueezeNet with Attention for Remote Sensing Scene Classification.
Azarbad, M., Azami, H., Sanei, S., Ebrahimzadeh, A., 2014. New Neural Network-based Approaches for GPS GDOP Classification based on Neuro-Fuzzy Inference System, Radial Basis Function, and Improved Bee Algorithm. Appl. Soft Comput. 25, 285–292. https://doi.org/10.1016/j.asoc.2014.09.022
Ba, L.J., Caruana, R., 2013. Do Deep Nets Really Need to be Deep?
Baker, B., Gupta, O., Naik, N., Raskar, R., 2017. Designing Neural Network Architectures using Reinforcement Learning. ArXiv161102167 Cs.
Baranwal, P., Batta, K., Kaushik, T., 2018. Comparative Study of Classical and Fuzzy PID Attitude Control System with Extended Kalman Filter Feedback for Nanosatellites.
Bazzi, H., Ienco, D., Baghdadi, N., Zribi, M., Demarez, V., 2020. Distilling Before Refine: Spatio-Temporal Transfer Learning for Mapping Irrigated Areas Using Sentinel-1 Time Series. IEEE Geosci. Remote Sens. Lett. 17, 1909–1913. https://doi.org/10.1109/LGRS.2019.2960625
Belward, A.S., Skøien, J.O., 2015. Who launched what, when and why; trends in global land-cover observation capacity from civilian earth observation satellites. ISPRS J. Photogramm. Remote Sens., Global Land Cover Mapping and Monitoring 103, 115–128. https://doi.org/10.1016/j.isprsjprs.2014.03.009
Blalock, D., Ortiz, J.J.G., Frankle, J., Guttag, J., 2020. What is the State of Neural Network Pruning? ArXiv200303033 Cs Stat.
Bonnet, J., Gleizes, M.-P., Kaddoum, E., Rainjonneau, S., Flandin, G., 2015. Multi-satellite Mission Planning Using a Self-Adaptive Multi-agent System, in: 2015 IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems. IEEE, Cambridge, MA, USA, pp. 11–20. https://doi.org/10.1109/SASO.2015.9
Browne, D., Giering, M., Prestwich, S., 2020. PulseNetOne: Fast Unsupervised Pruning of Convolutional Neural Networks for Remote Sensing. Remote Sens. 12, 1092. https://doi.org/10.3390/rs12071092
Bruhn, F.C., Tsog, N., Kunkel, F., Flordal, O., 2020. Enabling radiation tolerant heterogeneous GPU-based onboard data processing in space. Vol01234567891 3CEAS Space Journa 12, 551–564.
Buonaiuto, N., Kief, C., Louie, M., Aarestad, J., Zufelt, B., Mital, R., Mateik, D., Sivilli, R., Bhopale, A., 2017. Satellite Identification Imaging for Small Satellites Using NVIDIA 12.
Cai, Y., Hu, Y., Siegel, M., Gollapalli, S.J., Venugopal, A.R., Bardak, U., 2003. Onboard Feature Indexing from Satellite Lidar Images 4.
Cappi, C., Chapdelaine, C., Gardes, L., Jenn, E., Lefevre, B., Picard, S., Soumarmon, T., 2021. Dataset Definition Standard (DDS). ArXiv210103020 Cs.
Castelluccio, M., Poggi, G., Sansone, C., Verdoliva, L., 2015. Land Use Classification in Remote Sensing Images by Convolutional Neural Networks. ArXiv150800092 Cs.
CCSDS, Lossless Data Compression, 2020. 45.
CCSDS, Image Data Compression, 2021. 133.
Chai, Y., Fu, K., Sun, X., Diao, W., Yan, Z., Feng, Y., Wang, L., 2020. Compact Cloud Detection with Bidirectional Self-Attention Knowledge Distillation. Remote Sens. 12, 2770. https://doi.org/10.3390/rs12172770
Chan, M., Scarafoni, D., Duarte, R., Thornton, J., Skelly, L., 2018. Learning Network Architectures of Deep CNNs Under Resource Constraints, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Salt Lake City, UT, USA, pp. 1784–17847. https://doi.org/10.1109/CVPRW.2018.00222
Chen, G., Zhang, X., Tan, X., Cheng, Y., Dai, F., Zhu, K., Gong, Y., Wang, Q., 2018. Training Small Networks for Scene Classification of Remote Sensing Images via Knowledge Distillation. Remote Sens. 10, 719. https://doi.org/10.3390/rs10050719
Chen, H., Zhang, H., Boning, D., Hsieh, C.-J., 2019. Robust Decision Trees Against Adversarial Examples. ArXiv190210660 Cs Stat.
Chen, Y., Bian, Y., Xiao, X., Rong, Y., Xu, T., Huang, J., 2021. On Self-Distilling Graph Neural Network. ArXiv201102255 Cs Stat.
Cheng, C.-H., Shu, S.-L., Cheng, P.-J., 2009. Attitude

control of a satellite using fuzzy controllers. Expert Syst. Appl. 36, 6613–6620. https://doi.org/10.1016/j.eswa.2008.08.053

Cheng, C.-H., Yan, R., 2021. Testing Autonomous Systems with Believed Equivalence Refinement. ArXiv210304578 Cs.

Chien, S., Doubleday, J., Thompson, D.R., Wagstaff, K.L., Bellardo, J., Francis, C., Baumgarten, E., Williams, A., Yee, E., Stanton, E., Piug-Suari, J., 2017. Onboard Autonomy on the Intelligent Payload Experiment CubeSat Mission. J. Aerosp. Inf. Syst. 14, 307–315. https://doi.org/10.2514/1.I010386

cnes | Le site du Centre national d'études spatiales, 2022. URL https://cnes.fr/fr/ (accessed 7.18.22).

Codetta-Raiteri, D., Portinale, L., 2015. Dynamic Bayesian Networks for Fault Detection, Identification, and Recovery in Autonomous Spacecraft. IEEE Trans. Syst. Man Cybern. Syst. 45, 13–24. https://doi.org/10.1109/TSMC.2014.2323212

Cole, R.M., 2022. satellite-image-deep-learning.

Courbariaux, M., Bengio, Y., David, J.-P., 2015. BinaryConnect: Training Deep Neural Networks with binary weights during propagations, in: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (Eds.), Advances in Neural Information Processing Systems. Curran Associates, Inc.

D. Spiller, K. Thangavel, S. T. Sasidharan, S. Amici, L. Ansalone and R. Sabatini, "Wildfire segmentation analysis from edge computing for on-board real-time alerts using hyperspectral imagery," 2022 IEEE International Conference on Metrology for Extended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE), 2022, pp. 725-730, doi: 10.1109/MetroXRAINE54828.2022.9967553.

de Vieilleville, F., Lagrange, A., Ruiloba, R., May, S., 2020. Towards Distillation of Deep Neural Networks for Satellite On-Board Image Segmentation. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 43, 1553–1559.

Deepan, P., Sudha, L.R., 2020. Object Classification of Remote Sensing Image Using Deep Convolutional Neural Network, in: The Cognitive Approach in Cloud Computing and Internet of Things Technologies for Surveillance Tracking Systems. Elsevier, pp. 107–120. https://doi.org/10.1016/B978-0-12-816385-6.00008-8

Di, L., 2008. Standards, Critical Evaluation of Remote Sensing, in: Shekhar, S., Xiong, H. (Eds.), Encyclopedia of GIS. Springer US, Boston, MA, pp. 1128–1135. https://doi.org/10.1007/978-0-387-35973-1_1346

Del Rosso, M.P., Sebastianelli, A., Spiller, D., Mathieu, P.P., Ullo, S.L., 2021. On-Board Volcanic Eruption Detection through CNNs and Satellite Multispectral Imagery. Remote Sens. 13, 3479. https://doi.org/10.3390/rs13173479

Du, Y., Wang, T., Xin, B., Wang, L., Chen, Y., Xing, L., 2020. A Data-Driven Parallel Scheduling Approach for Multiple Agile Earth Observation Satellites. IEEE Trans. Evol. Comput. 24, 679–693. https://doi.org/10.1109/TEVC.2019.2934148

Elsken, T., Metzen, J.H., Hutter, F., 2019. Neural Architecture Search: A Survey. ArXiv180805377 Cs Stat.

Esposito, M., Conticello, S.S., Pastena, M., Domínguez, B.C., 2019. In-orbit demonstration of artificial intelligence applied to hyperspectral and thermal sensing from space, in: CubeSats and SmallSats for Remote Sensing III. International Society for Optics and Photonics, p. 111310C. https://doi.org/10.1117/12.2532262

Faisal D'Souza, 2019. The National Artificial Intelligence Research and Development Strategic Plan: 2019 Update 50.

Frankle, J., Dziugaite, G.K., Roy, D.M., Carbin, M., 2021. Pruning Neural Networks at Initialization: Why are We Missing the Mark? ArXiv200908576 Cs Stat.

Frédéric Férésin, Erwann Kervennic, Yves Bobichon, Edgar Lemaire, Nassim Abderrahmane, Gaétan Bahk, Ingrid Grenet, Matthieu Moretti, Michaël Benguigui, 2021. In space image processing using AI embedded on system on module : example of OPS-SAT cloud segmentation.

Fuertes, S., Pilastre, B., D'Escrivan, S., 2018. Performance assessment of NOSTRADAMUS & other machine learning-based telemetry monitoring systems on a spacecraft anomalies database, in: 2018 SpaceOps Conference. American Institute of Aeronautics and Astronautics, Marseille, France. https://doi.org/10.2514/6.2018-2559

Georges, L., Tanguy, S., Evridiki, N., David, E., 2021. In-Flight Training of a FDIR Model with Online Machine Learning on the OPS-SAT Spacecraft. URL https://github.com/georgeslabreche/opssat-orbitai/find/main (accessed 9.12.21).

Giuffrida, G., Diana, L., de Gioia, F., Benelli, G., Meoni, G., Donati, M., Fanucci, L., 2020.

CloudScout: A Deep Neural Network for On-Board Cloud Detection on Hyperspectral Images. Remote Sens. 12, 2205. https://doi.org/10.3390/rs12142205

Giuffrida, G., Fanucci, L., Meoni, G., Batič, M., Buckley, L., Dunne, A., van Dijk, C., Esposito, M., Hefele, J., Vercruyssen, N., Furano, G., Pastena, M., Aschbacher, J., 2022. The Φ-Sat-1 Mission: The First On-Board Deep Neural Network Demonstrator for Satellite Earth Observation. IEEE Trans. Geosci. Remote Sens. 60, 1–14. https://doi.org/10.1109/TGRS.2021.3125567

Globus, A., Crawford, J., Lohn, J., Pryor, A., 2003. Scheduling Earth Observing Satellites with Evolutionary Algorithms.

Goel, A., Tung, C., Lu, Y.-H., Thiruvathukal, G.K., 2020. A Survey of Methods for Low-Power Deep Learning and Computer Vision, in: 2020 IEEE 6th World Forum on Internet of Things (WF-IoT). Presented at the 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), pp. 1–6. https://doi.org/10.1109/WF-IoT48130.2020.9221198

Gong, Y., Liu, L., Yang, M., Bourdev, L., 2014. Compressing Deep Convolutional Networks using Vector Quantization. ArXiv14126115 Cs.

Goodfellow, I.J., Shlens, J., Szegedy, C., 2015. Explaining and Harnessing Adversarial Examples. ArXiv14126572 Cs Stat.

Goodwill, J., Crum, G., MacKinnon, J., Brewer, C., Monaghan, M., Wise, T., Wilson, C., 2021. NASA SpaceCube Edge TPU SmallSat Card for Autonomous Operations and Onboard Science-Data Analysis 13.

Goodwill, J., Wilson, D., Sabogal, S., George, A.D., Wilson, C., 2020. Adaptively Lossy Image Compression for Onboard Processing, in: 2020 IEEE Aerospace Conference. pp. 1–15. https://doi.org/10.1109/AERO47225.2020.9172536

Gregory Flandin, 2021. White Paper Machine Learning in Certified System 113.

Guo, Q., Fu, B., Shi, P., Cudahy, T., Zhang, J., Xu, H., 2017. Satellite Monitoring the Spatial-Temporal Dynamics of Desertification in Response to Climate Change and Human Activities across the Ordos Plateau, China. Remote Sens. 9, 525. https://doi.org/10.3390/rs9060525

Guo, Y., 2018. A Survey on Methods and Theories of Quantized Neural Networks. ArXiv180804752 Cs Stat.

Hadj-Salah, A., Guerra, J., Picard, M., Capelle, M., 2020. Towards operational application of Deep Reinforcement Learning to Earth Observation satellite scheduling.

Hadj-Salah, A., Verdier, R., Caron, C., Picard, M., Capelle, M., 2019. Schedule Earth Observation satellites with Deep Reinforcement Learning. ArXiv191105696 Cs.

Haikel, A., 2018. Multitask Classification of Remote Sensing Scenes Using Deep Neural Networks. Spain.

Han, H., Lee, S., Im, J., Kim, M., Lee, M.-I., Ahn, M.H., Chung, S.-R., 2015. Detection of Convective Initiation Using Meteorological Imager Onboard Communication, Ocean, and Meteorological Satellite Based on Machine Learning Approaches. Remote Sens. 7, 9184–9204. https://doi.org/10.3390/rs70709184

Han, S., Mao, H., Dally, W.J., 2016. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. ArXiv151000149 Cs.

He, T., Fan, Y., Qian, Y., Tan, T., Yu, K., 2014. Reshaping deep neural network for fast decoding by node-pruning, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Florence, Italy, pp. 245–249. https://doi.org/10.1109/ICASSP.2014.6853595

He, X., Zhao, K., Chu, X., 2021. AutoML: A survey of the state-of-the-art. Knowl.-Based Syst. 212, 106622. https://doi.org/10.1016/j.knosys.2020.106622

Hernández-Gómez, J.J., Yañez-Casas, G.A., Torres-Lara, A.M., Couder-Castañeda, C., Orozco-del-Castillo, M.G., Valdiviezo-Navarro, J.C., Medina, I., Solís-Santomé, A., Vázquez-Álvarez, D., Chávez-López, P.I., 2019. Conceptual low-cost on-board high performance computing in CubeSat nanosatellites for pattern recognition in Earth's remote sensing. pp. 114–104. https://doi.org/10.29007/8d25

Hinz, R., Bravo, J.I., Kerr, M., Marcos, C., Latorre, A., Membibre, F., 2020. EO-ALERT: Machine Learning-Based On-Board Satellite Processing for Very-Low Latency Convective Storm Nowcasting 1.

Hoeser, T., Bachofer, F., Kuenzer, C., 2020. Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications. Remote Sens. 12, 3053. https://doi.org/10.3390/rs12183053

Home,2022. Mindseed. URL https://www.mindseed.ie/ (accessed 7.18.22).

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. ArXiv170404861 Cs.

Huq, R., Bappy, M., Siddique, S., 2018. AI-OBC: Conceptual Design of a Deep Neural Network based Next Generation Onboard Computing Architecture for Satellite Systems.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. ArXiv160207360 Cs.

Ireland, M., 2019. Integrating AI Techniques Into Future Nanosatellite Onboard Data Processing 30.

ISO - ISO/IEC JTC 1/SC 42 - Artificial intelligence 2017. URL, https://www.iso.org/committee/6794475/x/catalogue/p/1/u/1/w/0/d/0 (accessed 9.4.21).

Ivano Verzola, Alessandro Donati, Martínez Heras, J.-A., Schubert, M., Laszlo Somodi, 2016. Project Sybil : A Novelty Detection System for Human Spaceflight Operations, in : Proc. Int. Conf. Space Operations.

Iverson, D.L., 2008. System Health Monitoring for Space Mission Operations, in: 2008 IEEE Aerospace Conference. IEEE, Big Sky, MT, USA, pp. 1–8. https://doi.org/10.1109/AERO.2008.4526646

Izzo, D., Märtens, M., Pan, B., 2018. A Survey on Artificial Intelligence Trends in Spacecraft Guidance Dynamics and Control. ArXiv181202948 Cs.

Izzo, D., Öztürk, E., 2021. Real-Time Guidance for Low-Thrust Transfers Using Deep Neural Networks. J. Guid. Control Dyn. 44, 315–327. https://doi.org/10.2514/1.G005254

Jalilian, S., SalarKaleji, F., Kazimov, T., 2017. Fault detection, isolation and recovery (FDIR) in satellite onboard software. https://doi.org/10.25045/NCSoftEng.2017.87

Jaekel, S., Scholz, B., 2015. Utilizing Artificial Intelligence to achieve a robust architecture for future robotic spacecraft, in: 2015 IEEE Aerospace Conference. IEEE, Big Sky, MT, pp. 1–14. https://doi.org/10.1109/AERO.2015.7119180

Joghataie, A., 1994. Neural Networks and Fuzzy Logic for Structural Control. University of Illinois Engineering Experiment Station. College of Engineering. University of Illinois at Urbana-Champaign.

Katz, G., Barrett, C., Dill, D., Julian, K., Kochenderfer, M., 2017. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. ArXiv170201135 Cs.

Kavzoglu, T., Mather, P.M., 1999. Pruning artificial neural networks: An example using land cover classification of multi-sensor images. Int. J. Remote Sens. 20, 2787–2803. https://doi.org/10.1080/014311699211796

Kim, S.-W., Park, S.-Y., Park, C., 2016. Spacecraft attitude control using neuro-fuzzy approximation of the optimal controllers. Adv. Space Res. 57, 137–152. https://doi.org/10.1016/j.asr.2015.09.016

Koizumi, S., Kikuya, Y., Sasaki, K., Masuda, Y., Iwasaki, Y., Watanabe, K., Yatsu, Y., Matsunaga, S., 2018. Development of Attitude Sensor using Deep Learning 8.

Kothari, V., Liberis, E., Lane, N.D., 2020. The Final Frontier: Deep Learning in Space. ArXiv200110362 Cs Eess.

Kucik, A., Meoni, G., 2021. Investigating Spiking Neural Networks for Energy-Efficient On-Board AI Applications. A Case Study in Land Cover and Land Use Classification. https://doi.org/10.1109/CVPRW53098.2021.00230

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks, in: Advances in Neural Information Processing Systems. Curran Associates, Inc.

Lane, N.D., Bhattacharya, S., Georgiev, P., Forlivesi, C., Kawsar, F., 2015. An Early Resource Characterization of Deep Learning on Wearables, Smartphones and Internet-of-Things Devices, in: Proceedings of the 2015 International Workshop on Internet of Things towards Applications. ACM, Seoul South Korea, pp. 7–12. https://doi.org/10.1145/2820975.2820980

Larq | Binarized Neural Network development, 2022. URL https://larq.dev/ (accessed 7.28.21).

Larry R, Davis, Kristin A. Fishburn, Helmut Lestinsky, Laurence R. Moore, Jennifer L. Walter, 2019. US Topo Product Standard (Techniques and Methods), Techniques and Methods.

Lazarevic, A., Obradovic, Z., 2001. Effective pruning of neural network classifier ensembles, in: IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222). IEEE, Washington, DC, USA, pp. 796–801. https://doi.org/10.1109/IJCNN.2001.939461

Li, J., Li, J., Chen, H., Jing, N., 2014. A data transmission scheduling algorithm for rapid-response earth-observing operations. Chin. J. Aeronaut. 27, 349–364. https://doi.org/10.1016/j.cja.2014.02.014

Liang, W., Li, J., Diao, W., Sun, X., Fu, K., Wu, Y., 2020. FGATR-Net: Automatic Network Architecture Design for Fine-Grained Aircraft Type Recognition in Remote Sensing Images. Remote Sens. 12, 4187. https://doi.org/10.3390/rs12244187

Linardatos, P., Papastefanopoulos, V., Kotsiantis, S., 2021. Explainable AI: A Review of Machine Learning Interpretability Methods. Entropy 23, 18. https://doi.org/10.3390/e23010018

Liping Di, Ben Kobler, 2000. NASA Standards for Earth Remote Sensing Data, URL https://www.researchgate.net/publication/228953572_NASA_Standards_for_Earth_Remote_Sensing_Data (accessed 9.4.21).

Liu, R., Yang, B., Zio, E., Chen, X., 2018. Artificial intelligence for fault diagnosis of rotating machinery: A review. Mech. Syst. Signal Process. 108, 33–47. https://doi.org/10.1016/j.ymssp.2018.02.016

Liu, X., 2020. Mission schedule of agile satellites based on Proximal Policy Optimization Algorithm. ArXiv200702352 Cs.

Liu, Yuchen, et al. “Mission Planning for Earth Observation Satellite With Competitive Learning Strategy.” Aerospace Science and Technology, vol. 118, Elsevier BV, Nov. 2021, p. 107047. Crossref, https://doi.org/10.1016/j.ast.2021.107047.

Ma, N., Yu, X., Peng, Y., Wang, S., 2019. A Lightweight Hyperspectral Image Anomaly Detector for Real-Time Mission. Remote Sens. 11, 1622. https://doi.org/10.3390/rs11131622

Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P., 2017. Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification. IEEE Trans. Geosci. Remote Sens. 55, 645–657. https://doi.org/10.1109/TGRS.2016.2612821

Mahajan, S., Fataniya, B., 2020. Cloud detection methodologies: variants and development—a review. Complex Intell. Syst. 6, 251–261. https://doi.org/10.1007/s40747-019-00128-0

Manning, J., Langerman, D., Ramesh, B., Gretok, E., Wilson, C., George, A., MacKinnon, J., Crum, G., 2018. Machine-Learning Space Applications on SmallSat Platforms with TensorFlow. Small Satell. Conf.

Mansour, M.A.A., Dessouky, M.M., 2010. A genetic algorithm approach for solving the daily photograph selection problem of the SPOT5 satellite. Comput. Ind. Eng. 58, 509–520. https://doi.org/10.1016/j.cie.2009.11.012

Maskey, A., Cho, M., 2020. CubeSatNet: Ultralight Convolutional Neural Network designed for on-orbit binary image classification on a 1U CubeSat. Eng. Appl. Artif. Intell. 96, 103952. https://doi.org/10.1016/j.engappai.2020.103952

Meß, J.-G., 2019. Techniques of Artificial Intelligence for Space Applications - A Survey.

Mirman, M., Gehr, T., Vechev, M., 2018. Differentiable Abstract Interpretation for Provably Robust Neural Networks, in: International Conference on Machine Learning. PMLR, pp. 3578–3586.

Mishra, A., Cook, J.J., Nurvitadhi, E., Marr, D., 2017. WRPN: Training and Inference using Wide Reduced-Precision Networks. ArXiv170403079 Cs.

Mittal, S., 2019. A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform. J. Syst. Archit. 97, 428–442. https://doi.org/10.1016/j.sysarc.2019.01.011

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing Atari with Deep Reinforcement Learning. ArXiv13125602 Cs.

Molchanov, D., Ashukha, A., Vetrov, D., 2017. Variational Dropout Sparsifies Deep Neural Networks. ArXiv170105369 Cs Stat.

Molnar, C., 2021. Interpretable Machine Learning.

Müller, M.N., Makarchuk, G., Singh, G., Püschel, M., Vechev, M., 2021. PRIMA: Precise and General Neural Network Certification via Multi-Neuron Convex Relaxations 20.

Nguyen, C.V., Achille, A., Lam, M., Hassner, T., Mahadevan, V., Soatto, S., 2019. Toward Understanding Catastrophic Forgetting in Continual Learning. ArXiv190801091 Cs Stat.

olanleed, 2021. *MochiMochi*. 2021. Accessed: Sep. 29, 2021. [Online]. Available: https://github.com/olanleed/MochiMochi.

Olive, X., 2010. FDI(R) for satellite at Thales Alenia Space how to deal with high availability and robustness in space domain?, in: 2010 Conference on Control and Fault-Tolerant Systems (SysTol). Presented at the 2010 Conference on Control and Fault-Tolerant Systems (SysTol), IEEE, Nice, pp. 837–842. https://doi.org/10.1109/SYSTOL.2010.5675942

O’Meara, C., Schlag, L., Faltenbacher, L., Wickler, M., 2016. ATHMoS: Automated Telemetry Health Monitoring System at GSOC using Outlier Detection and Supervised Machine Learning. https://doi.org/10.2514/6.2016-2347

Ortega, G., 1995. Fuzzy logic techniques for rendezvous and docking of two geostationary satellites. Telemat. Inform., Advanced Space Technologies For Systems Autonomy 12, 213–227. https://doi.org/10.1016/0736-5853(95)00013-5

Pan, G., Xu, Y., Ma, J., 2021. The potential of CO2 satellite monitoring for climate governance: A review. J. Environ. Manage. 277, 111423. https://doi.org/10.1016/j.jenvman.2020.111423

Pang, T., Xu, K., Du, C., Chen, N., Zhu, J., 2019. Improving Adversarial Robustness via Promoting Ensemble Diversity, in: Proceedings of the 36th International Conference on Machine Learning. PMLR, pp. 4970–4979.

Pant, Ayush. "Workflow of a Machine Learning Project." Medium, 23 Jan. 2019, towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94.

Peng, S., Chen, H., Du, C., Li, J., Jing, N., 2018. Onboard Observation Task Planning for an Autonomous Earth Observation Satellite Using Long Short-Term Memory. IEEE Access 6, 65118–65129. https://doi.org/10.1109/ACCESS.2018.2877687

Pilastre, B., 2020. Estimation parcimonieuse et apprentissage de dictionnaires pour la détection d'anomalies multivariées dans des données mixtes de télémesure satellites (phd).

Pitsis, G., Tsagkatakis, G., Kozanitis, C., Kalomoiris, I., Ioannou, A., Dollas, A., Katevenis, M.G.H., Tsakalides, P., 2019. Efficient Convolutional Neural Network Weight Compression for Space Data Classification on Multi-fpga Platforms, in: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Brighton, United Kingdom, pp. 3917–3921. https://doi.org/10.1109/ICASSP.2019.8682732

Polino, A., Pascanu, R., Alistarh, D., 2018. Model compression via distillation and quantization. ArXiv180205668 Cs.

Poortinga, A., Thwal, N.S., Khanal, N., Mayer, T., Bhandari, B., Markert, K., Nicolau, A.P., Dilger, J., Tenneson, K., Clinton, N., Saah, D., 2021. Mapping sugarcane in Thailand using transfer learning, a lightweight convolutional neural network, NICFI high resolution satellite imagery and Google Earth Engine. ISPRS Open J. Photogramm. Remote Sens. 1, 100003. https://doi.org/10.1016/j.ophoto.2021.100003

Post-training quantization | TensorFlow Lite, 2022. URL https://www.tensorflow.org/lite/performance/post_training_quantization (accessed 9.28.21).

Pruning in Keras example | TensorFlow Model Optimization , 2022. . TensorFlow. URL https://www.tensorflow.org/model_optimization/guide/pruning/pruning_with_keras (accessed 8.6.21).

Qin, H., Gong, R., Liu, X., Bai, X., Song, J., Sebe, N., 2020. Binary Neural Networks: A Survey. Pattern Recognit. 105, 107281. https://doi.org/10.1016/j.patcog.2020.107281

Quantization — PyTorch 1.9.1 documentation, 2022. URL https://pytorch.org/docs/stable/quantization.html (accessed 7.28.21).

Ranasinghe, K., Sabatini, R., Gardi, A., Bijjahalli, S., Kapoor, R., Fahey, T., Thangavel, K., 2022. Advances in Integrated System Health Management for Mission-essential and Safety-critical Aerospace Applications. Prog. Aerosp. Sci. 128, 100758. https://doi.org/10.1016/j.paerosci.2021.100758

Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A., 2016. XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks. ArXiv160305279 Cs.

Ricard Castaño, Steve Ankuo Chien, Kiri L. Wagstaff, Timothy M. Stough, 2007. On-board analysis of uncalibrated data for a spacecraft at mars, in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, August 12-15, 2007. San Jose, California, USA. https://doi.org/10.1145/1281192.1281291

Ricks, B.W. and Mengshoel, O.J., 2009. Methods for probabilistic fault diagnosis: An electrical power system case study. In Annual Conference of the PHM Society (Vol. 1, No. 1).

Rieke, C., 2022. Awesome Satellite Imagery Datasets. Github: https://github.com/chrieke/awesome-satellite-imagery-datasets

SatSure, 2022. URL: https://satsure.co/ (accessed 7.18.22).

Schumann, J., Mengshoel, O.J., Mbaya, T., 2011. Integrated Software and Sensor Health Management for Small Spacecraft, in: 2011 IEEE Fourth International Conference on Space Mission Challenges for Information Technology. IEEE, Palo Alto, CA, USA, pp. 77–84. https://doi.org/10.1109/SMC-

IT.2011.25
Shalev-Shwartz, S., 2014. Understanding Machine Learning: From Theory to Algorithms, 1st edition. ed. Cambridge University Press, New York, NY, USA.
Shaw, G.A., Burke, H.K., 2003. Spectral Imaging for Remote Sensing 14, 26.
Shweta, K., 2019. A Survey on Classification of Concept Drift with Stream Data.
Simons, T., Lee, D.-J., 2019. A Review of Binarized Neural Networks. Electronics 8, 661. https://doi.org/10.3390/electronics8060661
Simonyan, K., Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. ArXiv14091556 Cs.
Song, Y., Zhou, Z., Zhang, Z., Yao, F., Chen, Y., 2020. A framework involving MEC: imaging satellites mission planning. Neural Comput. Appl. 32, 15329–15340. https://doi.org/10.1007/s00521-019-04047-6
Srivastava, A.N., 2003. Onboard Detection of Snow, Ice, Clouds and Other Geophysical Processes Using Kernel Methods.
Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V., 2019. MnasNet: Platform-Aware Neural Architecture Search for Mobile, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Long Beach, CA, USA, pp. 2815–2823. https://doi.org/10.1109/CVPR.2019.00293
Tan, Q., Ling, J., Hu, Jun, Qin, X., Hu, Jiping, 2020. Vehicle Detection in High Resolution Satellite Remote Sensing Images Based on Deep Learning. IEEE Access 8, 153394–153402. https://doi.org/10.1109/ACCESS.2020.3017894
Tang, Y., Ji, J., Gao, S., Dai, H., Yu, Y., Todo, Y., 2018. A Pruning Neural Network Model in Credit Classification Analysis. Comput. Intell. Neurosci. 2018, 1–22. https://doi.org/10.1155/2018/9390410
Tjoa, E., Guan, C., 2020. A Survey on Explainable Artificial Intelligence (XAI): Towards Medical XAI. IEEE Trans. Neural Netw. Learn. Syst. 1–21. https://doi.org/10.1109/TNNLS.2020.3027314
Törnblom, J., Nadjm-Tehrani, S., 2019. Formal Verification of Random Forests in Safety-Critical Applications, in: Artho, C., Ölveczky, P.C. (Eds.), Formal Techniques for Safety-Critical Systems, Communications in Computer and Information Science. Springer International Publishing, Cham, pp. 55–71. https://doi.org/10.1007/978-3-030-12988-0_4
Thangavel, K., Spiller, D., Sabatini, R., Amici, S., Sasidharan, S.T., Fayek, H. and Marzocca, P., 2023. Autonomous Satellite Wildfire Detection Using Hyperspectral Imagery and Neural Networks: A Case Study on Australian Wildfire. Remote Sensing, 15(3), p.720.
Thangavel, K.; Spiller, D.; Sabatini, R.; Marzocca, P., 2022. On-board Data Processing of Earth Observation Data Using 1-D CNN. SmartSat CRC Conference, New South Wales, Australia, 12–13 September 2022. DOI: 10.13140/RG.2.2.16042.70088.
Thangavel, K., Spiller, D., Sabatini, R., Amici, S., Sasidharan, S.T., Fayek, H. and Marzocca, P., 2023. Autonomous Satellite Wildfire Detection Using Hyperspectral Imagery and Neural Networks: A Case Study on Australian Wildfire. Remote Sensing, 15(3), p.720.
Vladimirova, T., Atek, S., 2002. A New Lossless Compression Method for Small Satellite On-Board Imaging. University of Surrey, University of Surrey Guildford, Surrey, GU2 7 XH United Kingdom. https://doi.org/10.1142/9789812776266_0038
Voss, S., 2019. Application of Deep Learning for Spacecraft Fault Detection and Isolation. Delft University of Technology.
Wagstaff, K.L., Altinok, A., Chien, S.A., Rebbapragada, U., Schaffer, S.R., Thompson, D.R., Tran, D.Q., 2017. Cloud Filtering and Novelty Detection using Onboard Machine Learning for the EO-1 Spacecraft. Int. Jt. Conf. Artif. Intell. 4.
Wahlster, W., Cristoph Winterhalter, 2020. GERMAN STANDARDIZATION ROADMAP ON ARTIFICIAL INTELLIGENCE 226.
Wang, B., 2021. Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX.
Wang, H., Qin, C., Zhang, Y., Fu, Y., 2021. Emerging Paradigms of Neural Network Pruning. ArXiv210306460 Cs.
Wang, H., Yang, Z., Zhou, W., 2019. Online scheduling of image satellites based on neural networks and deep reinforcement learning 32, 9.
Wang, S., Pei, K., Whitehouse, J., Yang, J., Jana, S., 2018. Formal Security Analysis of Neural Networks using Symbolic Intervals. ArXiv180410829 Cs.
Wang, X., Wu, G., Xing, L., Pedrycz, W., 2021. Agile Earth observation satellite scheduling over 20 years: formulations, methods and future directions. IEEE Syst. J. 15, 3881–3892. https://doi.org/10.1109/JSYST.2020.2997050
Wang, Y., Ma, Z., Yang, Y., Wang, Z., tang, L., 2019. A New Spacecraft Attitude Stabilization

Mechanism Using Deep Reinforcement Learning Method 13 pages. https://doi.org/10.13009/EUCASS2019-33
Wertz, J.R., Larson, W.J., 1999. Space Mission Analysis and Design, 3rd edition. ed. Springer, El Segundo, Calif.: Dordrecht; Boston.
White Paper on Artificial Intelligence: a European approach to excellence and trust, 2020. Eur. Comm. - Eur. Comm. URL https://ec.europa.eu/info/publications/white-paper-artificial-intelligence-european-approach-excellence-and-trust_en (accessed 9.11.21).
Winter, P.M., Eder, S.K., Weissenbock, J., Schwald, C., Doms, T., Vogt, T., Hochreiter, S., Nessler, B., 2021. Trusted Artificial Intelligence: Towards Certification of Machine Learning Applications. ArXiv abs/2103.16910.
Wu, S.-F., Engelen, C.J.H., Chu, Q.-P., Babuška, R., Mulder, J.A., Ortega, G., 2001. Fuzzy logic based attitude control of the spacecraft X-38 along a nominal re-entry trajectory. Control Eng. Pract. 9, 699–707. https://doi.org/10.1016/S0967-0661(01)00036-3
Yadava, D., Hosangadi, R., Krishna, S., Paliwal, P., Jain, A., 2018. Attitude control of a nanosatellite system using reinforcement learning and neural networks, in: 2018 IEEE Aerospace Conference. IEEE, Big Sky, MT, pp. 1–8. https://doi.org/10.1109/AERO.2018.8396409
Yang, Z., Li, L., Xu, X., Kailkhura, B., Xie, T., Li, B., 2021. On the Certified Robustness for Ensemble Models and Beyond. ArXiv210710873 Cs.
Yu, D., Xu, Q., Guo, H., Zhao, C., Lin, Y., Li, D., 2020. An Efficient and Lightweight Convolutional Neural Network for Remote Sensing Image Scene Classification. Sensors 20, 1999. https://doi.org/10.3390/s20071999
Zhang, B., Zhang, Y., Wang, S., 2019. A Lightweight and Discriminative Model for Remote Sensing Scene Classification With Multidilation Pooling Module. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 12, 2636–2653. https://doi.org/10.1109/JSTARS.2019.2919317
Zhang, S., Wu, G., Gu, J., Han, J., 2020. Pruning Convolutional Neural Networks with an Attention Mechanism for Remote Sensing Image Classification. Electronics 9, 1209. https://doi.org/10.3390/electronics9081209
Zhang, Z., Iwasaki, A., Xu, G., Song, J., 2019. Cloud detection on small satellites based on lightweight U-net and image compression. J. Appl. Remote Sens. 13, 026502. https://doi.org/10.1117/1.JRS.13.026502
Zhang Z, Li G, Xu Y, Tang X. Application of Artificial Intelligence in the MRI Classification Task of Human Brain Neurological and Psychiatric Diseases: A Scoping Review. Diagnostics. 2021; 11(8):1402. https://doi.org/10.3390/diagnostics11081402
Zhao, S., Yue, X., Zhang, S., Li, B., Zhao, H., Wu, B., Krishna, R., Gonzalez, J.E., Sangiovanni-Vincentelli, A.L., Seshia, S.A., Keutzer, K., 2020. A Review of Single-Source Deep Unsupervised Visual Domain Adaptation. ArXiv200900155 Cs Eess.
Zhu, C., Han, S., Mao, H., Dally, W.J., 2017. Trained Ternary Quantization. ArXiv161201064 Cs.
Zoph, B., Le, Q.V., 2017. Neural Architecture Search with Reinforcement Learning. ArXiv161101578 Cs.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.