# Edge computing for Earth Observation: challenges and current trends

Dr. Gabriele Meoni

Innovation Officer, ESA ESRIN Frascati, Rome, Italy

Email: gabriele.meoni@esa.int

# A brief introduction of me

- **Name**: Gabriele Meoni, PhD
- **Role**: Innovation Officer at European Space Agency
  - Bringing transformational innovation to Earth Observation applications
  - **Sections**:
    - Advanced Concepts and Studies Office
    - Φ-lab
  - IEEE GRSS Earth Science Informatics (ESI): technical committee co-chair
- **Topic of interests**:
  - Edge computing for space
  - Artificial Intelligence for Earth Observations
  - Quantum Computing


**Contacts**:

gabriele.meoni@esa.int

gabriele-meoni-53b390131

# Learning objectives

- At the end of this lesson, you will be able to:
  - Describe the concept of edge computing and its main applications in EO missions
  - Describe the main hardware devices for edge computing in space
  - Discuss the main technical challenges and describe related research trends

# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

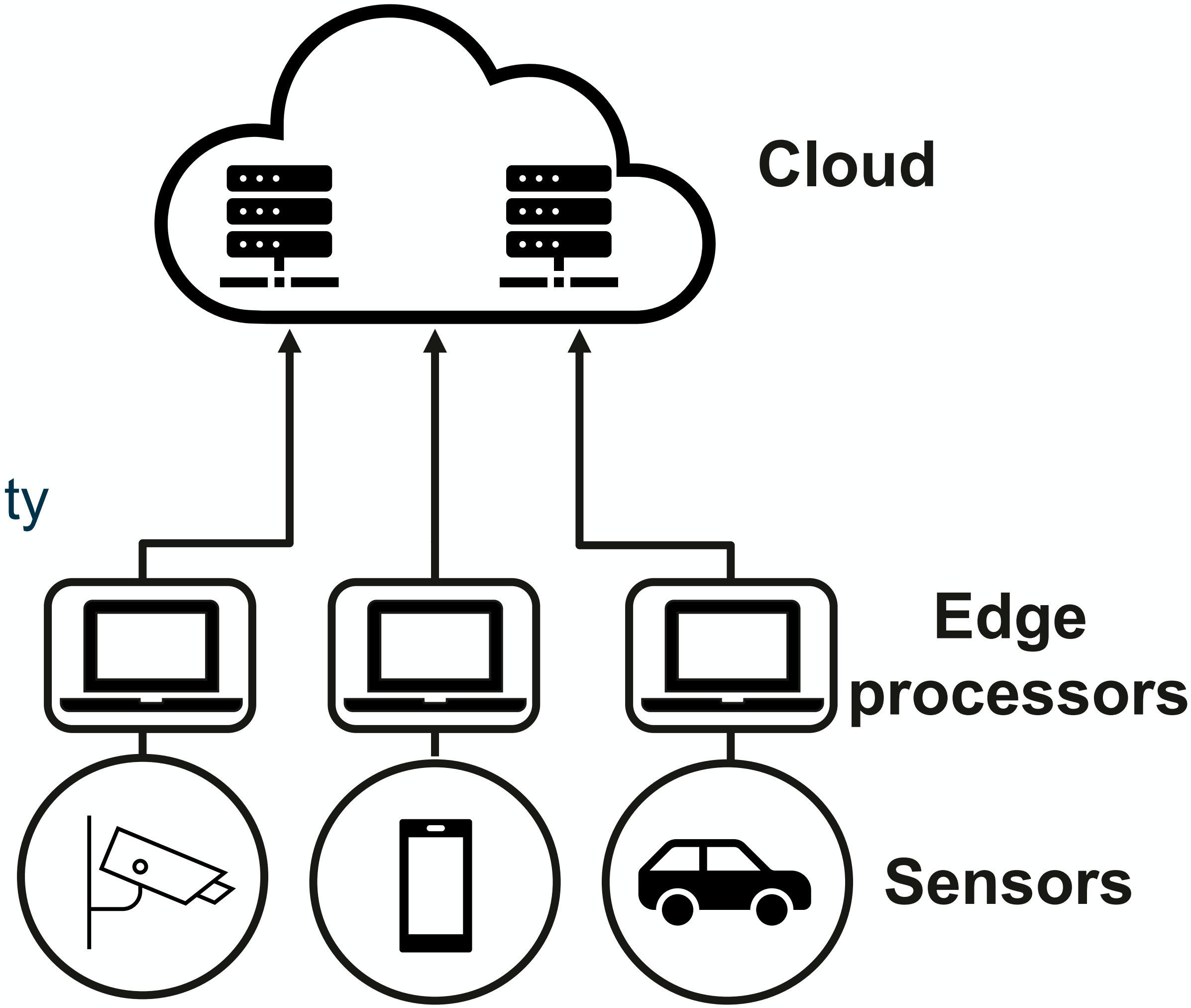
# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

# What is edge computing?

- **Cloud computing model**
  - Sending nodes producing data
  - **Cloud:** distributed computing infrastructure for data processing (e.g., data centres, servers)
  - **One limitation**: large amount of raw data distributed on the network. Limited scalability for highly growing networks

- **Edge computing model**
  - (Pre-)processing data at the ***edge*** of the network, i.e., close to the sensors
  - Transmitting only "pre-filtered" data to the cloud
  - Sensors are equipped through "edge" processors (or servers)



# What about space? The bent-pipe approach (1/2)

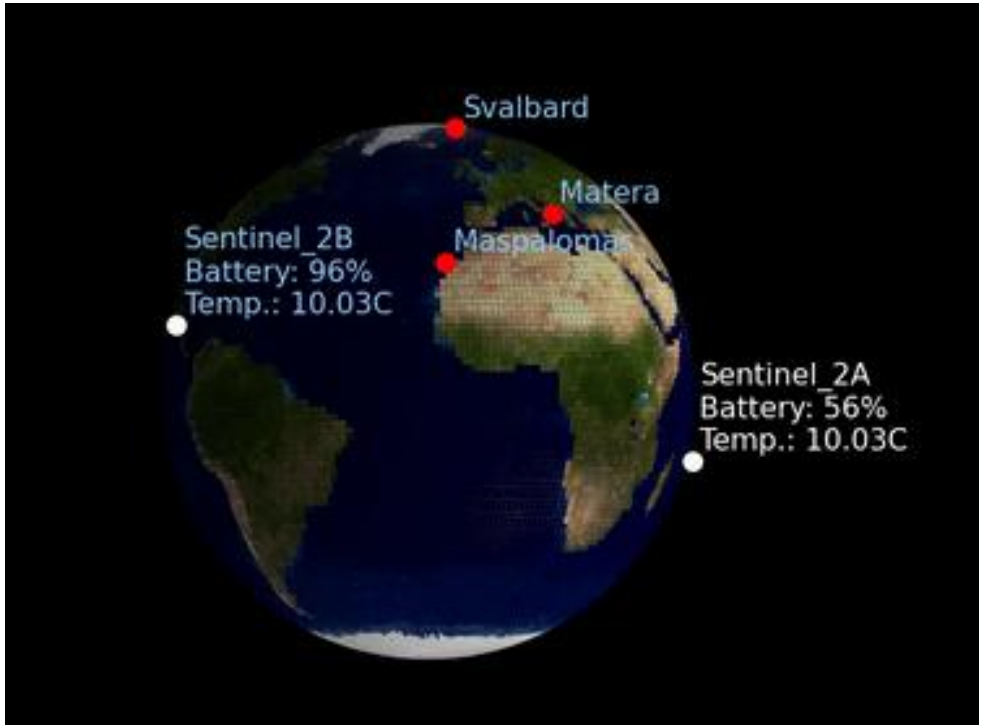



Image source: https://github.com/aidotse/PASEOS.

(Battery and temperature estimates not realistic)

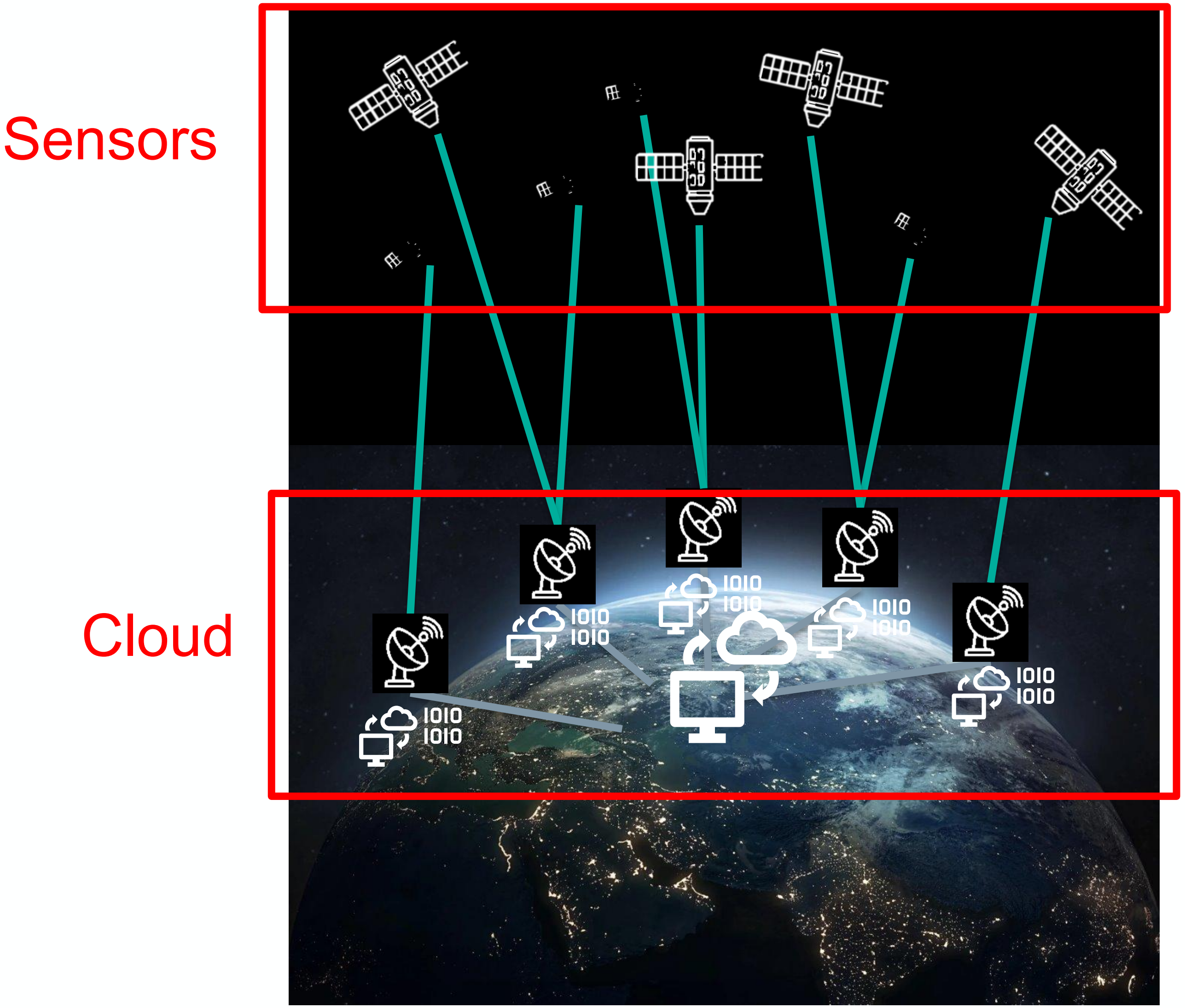
**Sentinel-2: orbital period**: 100.6 min

- **Bent-pipe**
  - Satellites receive a «command» when close to the ground station.
  - Satellite transmit data to the ground
  - Data are processed on the ground segment and then transferred to the end users

## Which limitations do you think the bent-pipe approach has?

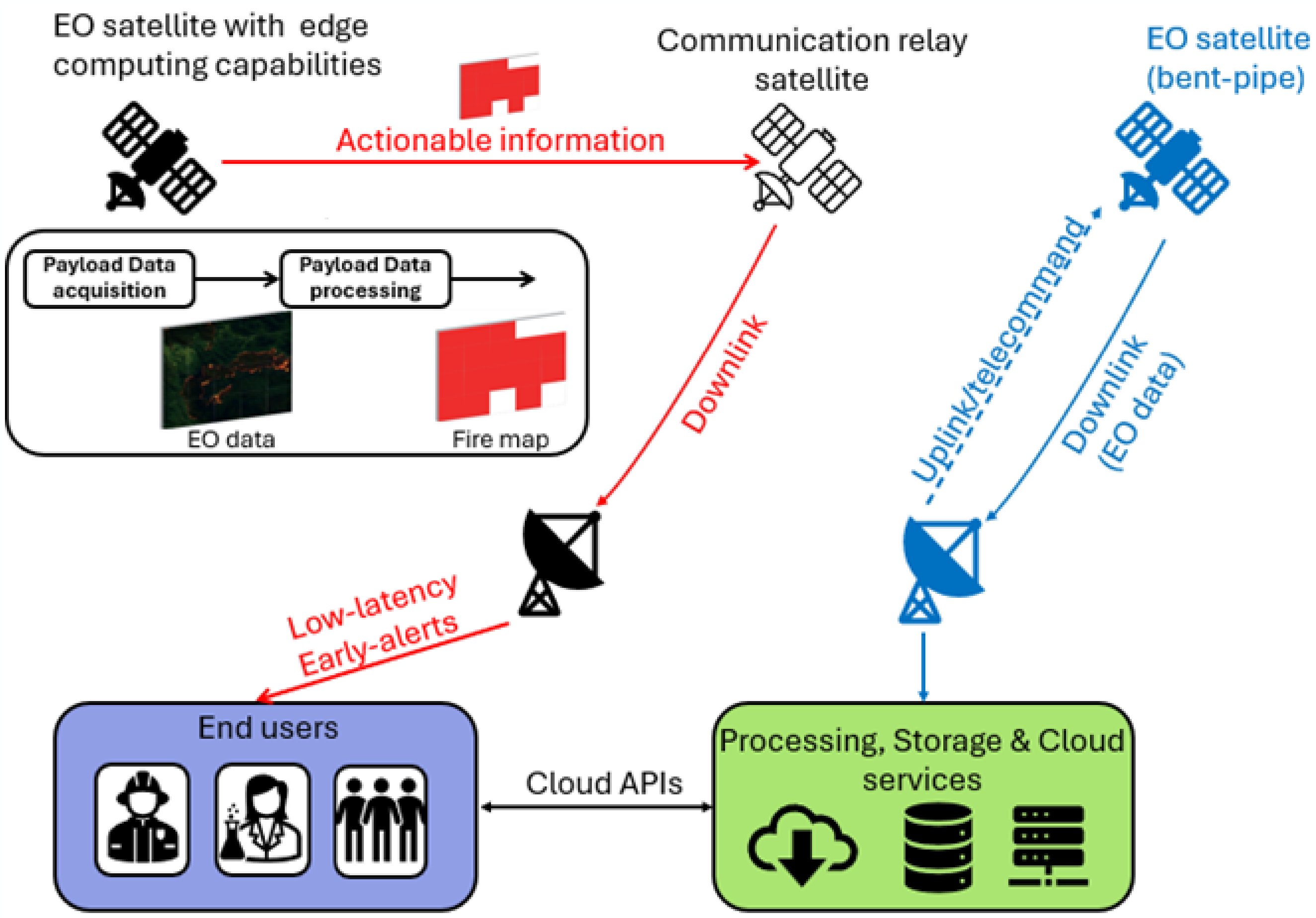
# What about space? The bent-pipe approach (2/2)

- **Bent pipe:** designed to transmit **vast amount of data** to end users
- **Limitations**: Not optimised for:
  - fast download of **actionable information** to the end users




How can we improve this via edge computing?

## Edge computing in space: processing data directly on board



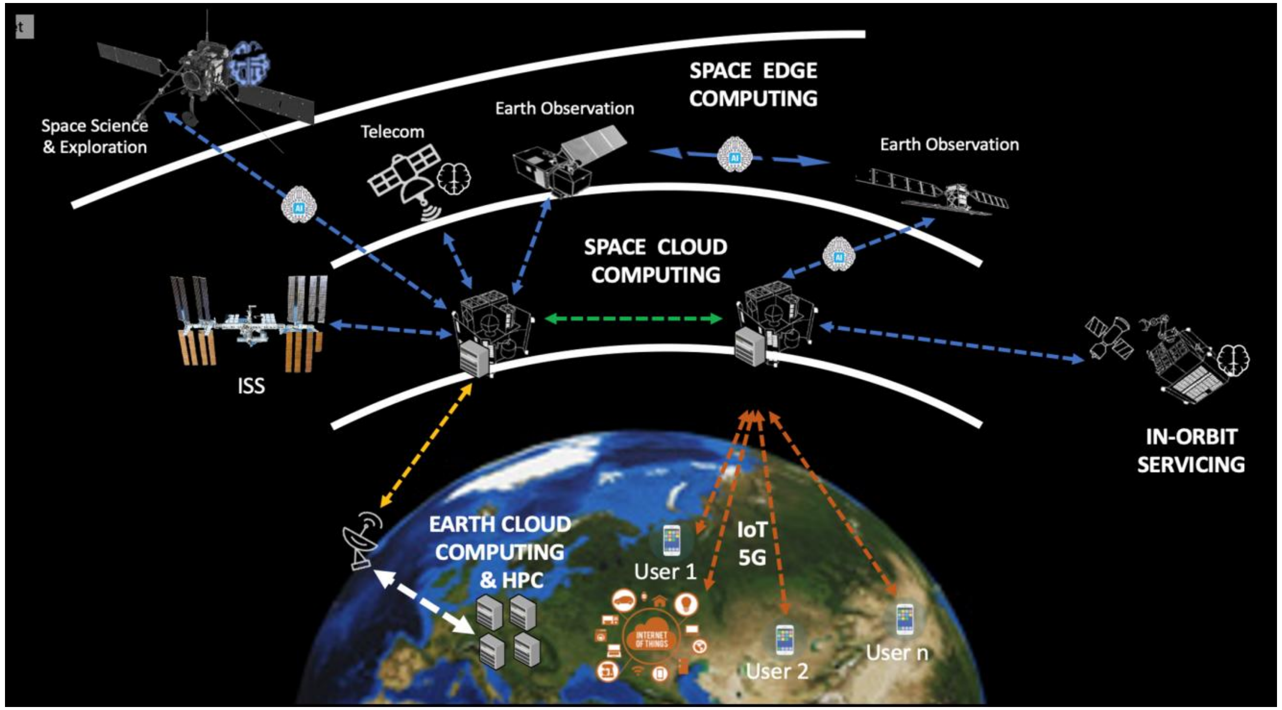

- **Artificial Intelligence (AI)** algorithms process EO data on board satellites to extract **actionable information** (e.g., fire alert + map) [1]
- Actionable information can be transmitted to end users quickly through a communication relay satellite
- **Why AI**:
  - State-of-the-art for many EO applications
  - AI hardware is versatile and reconfigurable
- **Advantages**
  - Higher autonomy and responsiviness
  - Easy reconfigurability
  - Data efficient (e.g., possibility to remove unwanted data)
- N.B. **It complements** bent-pipe (not-replace!)

# Cognitive Cloud Computing in Space (3CS)

esa Φ-lab

- **3CS:** integration of AI, novel computing paradigms (cloud computing, edge computing, etc), computing hardware inside space infrastructure to enhance its autonomy, agility and reconfigurability.

- Multiple in-orbit space assets connected (even non-EO satellites)




Courtesy: Letter to DG from 10 New Space Companies

- To create a clear technological vision, highlight novel business models and understand the value of 3CS in the next 10 years, ESA Φ-lab is now running 3 parallel projects

# Edge Computing in Space: where

- Satellites are composed by main 2 elements:
  - **Platform**
    - Responsible for main satellite functions (e.g., Power distribution, Attitude & Orbital Control, …)
    - The main computing element is called **Onboard Computer (OBC)**
  - One or more **Payloads**
    - Responsible for scientific/commercial purposes of the mission (e.g., EO data acquisition, processing,…)
    - The main computing element is called **Payload Data Handling Unit (PDHU)**

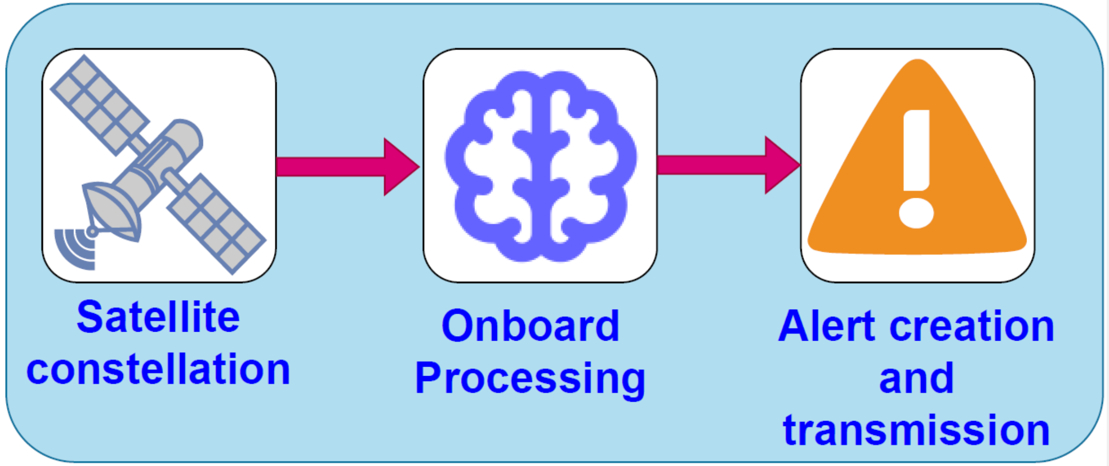
Edge computing relevant to both

We limit our exploration to edge computing for payload data processing

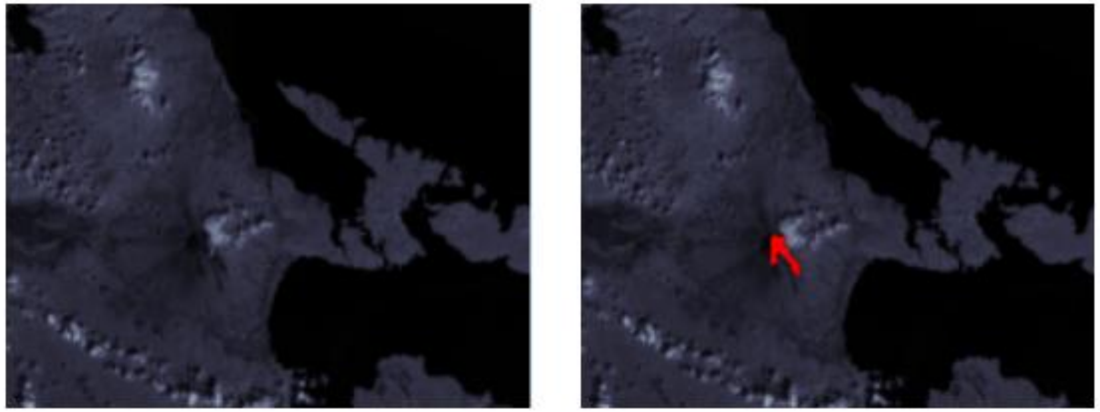
# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

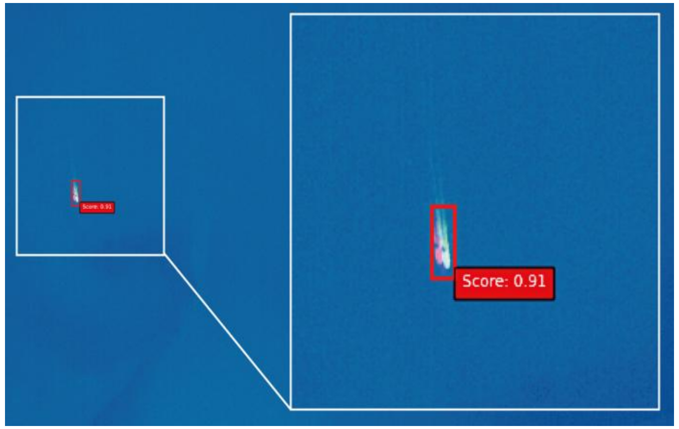
# Low-latency early-alert systems




**Volcanic eruption**



**Maritime surveillances**

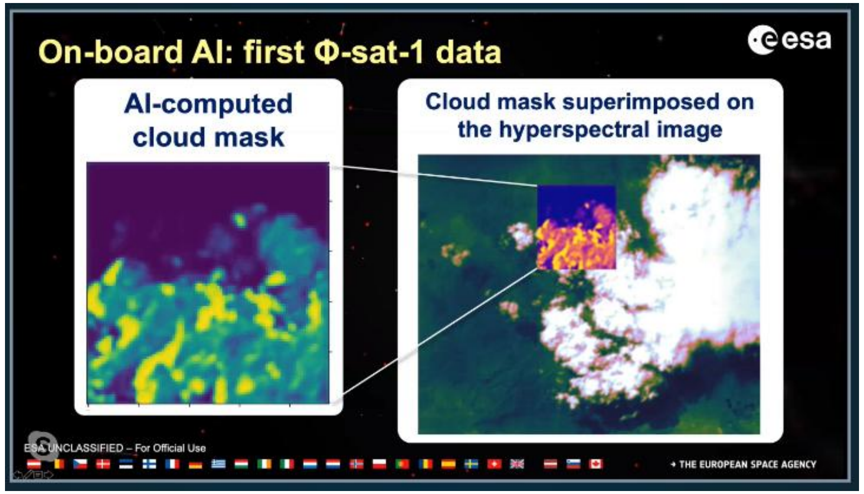



Credits: Dr Roberto Del Prete

- **Fast delivery** of actionable insights is promising for **civil-security** applications (disaster monotoring and early warning, maritime surveillance applications)

# Payload Data reduction/filtering

- **Clouds**:
  - 50% (or more) data are covered by clouds
  - Optical images suffer of cloud-occlusion
- By using AI on board satellites, you can:
  - detect cloudy images and discarding them before transmitting
  - Saving bandwidth for useful images
  - Demonstrated by Φ-Sat-1 mission
  - You can extend this paradigm to corrupted or non-wanted images.

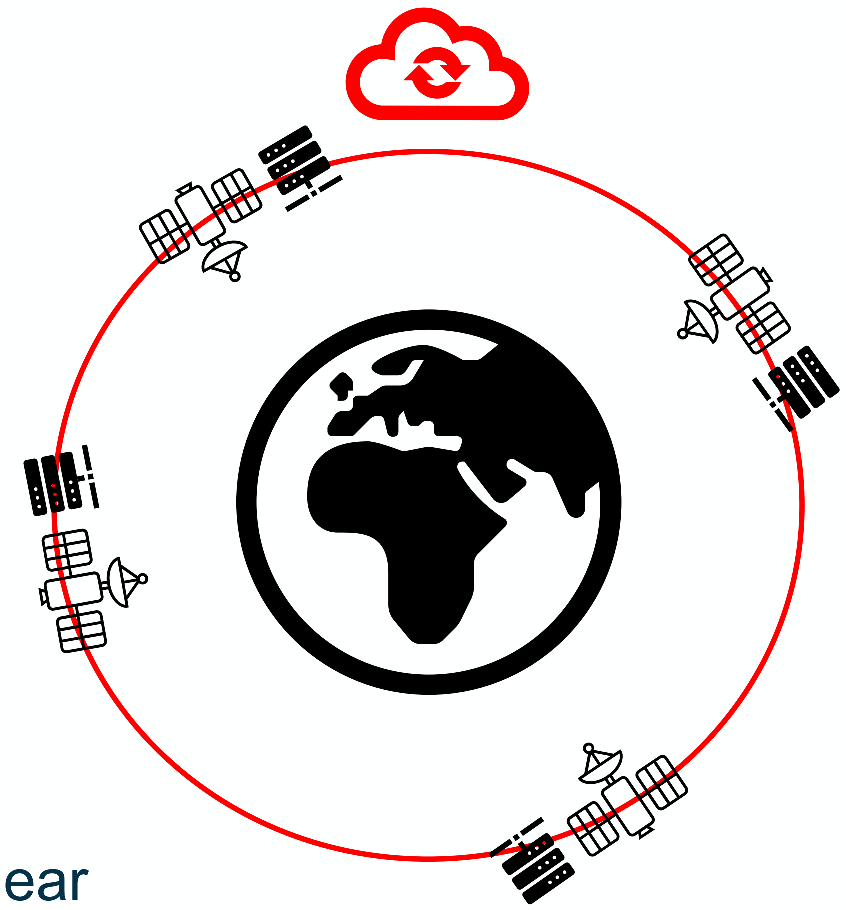



Credit: ESA

# Payload Data reduction/filtering

- In-orbit cloud computing (e.g., space-based data centres)

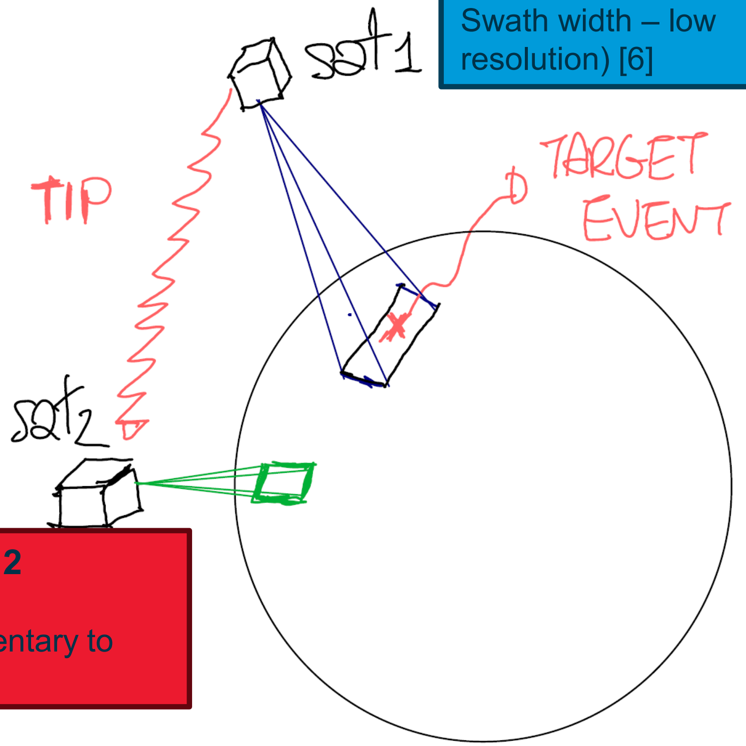
- Satellite platforms as service
  - Satellite heterogeneous constellations
  - You can task one/multiple satellites acquisitions
  - Data are processed in-orbit as apps in the cloud
  - Information extracted from data can be used to task future acquisitions
- Convenience compared to on ground data centres still not clear



# Novel acquisition systems (Tip & Cue)

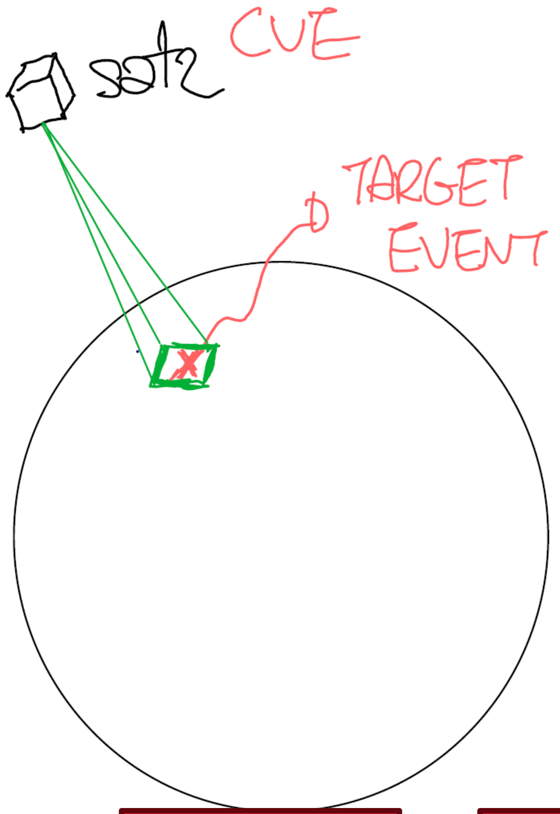
## TIP

**Satellite 1**
Sensor 1 (typically high Swath width – low resolution) [6]




**Satellite 2**
Sensor 2 complementary to sensor 1

## CUE




**Satellite 1 acquisition** → **Satellite 1** Onboard data processing → Tipping → **Satellite 2** acquisition planning → **Satellite 2 acquisition** → **Satellite 2** Onboard data processing

# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

# Metrics to evaluate hardware for payload data processing (1/4)

- **Computing power/performance**:
  - Typically measured through:
    - General Indicators: FLOPS or specific benchmarks (e.g., DHRYSTONE)
    - Application-specific indicators: latency (s), throughput (e.g., Frames Per Seconds)
  - **Why important**: some applications require fast responses => fast in-orbit processing

- **Energy-efficiency:**
  - Typically measured through:
    - General Indicators: [J], Energy-per-operation
    - Application-specific indicators: Energy/inference

# Metrics to evaluate hardware for payload data processing (2/4)

- **Energy-efficiency:**
  - **Why important**:
    - Limited energy availability
    - Thermal constraints: energy dissipated as heat might lead to increase satellite internal temperature
    - **Question**: what is the difference between power consumption and energy consumption?
      - $E = P \cdot \Delta T$

- **Radiation:**
  - Three main phenomena:
    - **Single Event Effects**: incoming charged particles induce glitches in digital electronics which might induce (temporary) faults (which can be solved by resetting the system)
    - **Total Ionization Dose**: accumulated charge due trapping of incoming incoming particles. In the long term, it permanently switches on/off transistors leading to permanent failure (typically measured in RAD)
    - **Single Event Latch-ups**: high-energy impacting particles triggers positive feedback effects causing permanent failures
  - **Why important**: decreasing systems reliability

# Metrics to evaluate hardware for payload

- **Cost:**
  - Typically measured in: €
  - Cost can be divided into two components:
    - $C\_NRE$ (Non-recurring engineering cost). Non depending on the number on number of pieces produced ($N$)
    - $C\_RE$ (Recurring engineering cost). $C\_RE \alpha N$

- **Size, mass:**
  - Typically measured in: m$^3$, kg
  - **Why important**: impacting design, costs

# Microcontrollers

- **How do they work**
  - Fixed hardware executing a software

- **General Purpose** (same hardware for all the applications)
  - Flexible (new software => new application)
  - Low-performance/ low energy efficiency (hardware not optimized for any application)
  - Low-development time (just need to develop the software) ($C_{NRE}$ low, $C_{RE}$ low )
  - Radiation tolerance depending on design and technology

- Typically used in CubeSats or satellites with very limited processing capabilities

# Field Programmable Gate Arrays (1/2)

- FPGAs including **Programmable hardware**:
  - Programmable logic and connections
  - Algorithms are mapped in **hardware**
    - The designer writes the description of the hardware creating a specific hardware for an application

- System on a Chip FPGAS (**SoC FPGAs**) - FPGAs coupled with processors
  - Some algorithms are more performant when implemented in software
  - FPGAs are convenient for parallel data-flow algorithms

# Field Programmable Gate Arrays (2/2)

- Limited flexibility (new application = hardware redesign)
- High-performance – Pretty energy-efficient
  - Typically used for applications requiring high-data-rate processing (e.g., DNN inference, SAR data processing/focusing)
- High development time: need to design an entire hardware ($C_{NRE}$ : high, $C_{RE}$: higher than microcontrollers)
- Radiation tolerance depending on design and technology
- **High-level-Synthesis**: possibility to synthesize hardware from Python or other high-level languages (faster implementation, lower performance)


OPS-SAT (PDHU): MitySOM-5CSX SoM (Copyright: ESA)

# Edge Graphics Processing Units

- **Graphics Processing Units (GPUs):**
  - Fixed and massively parallel hardware optimized for data-flow processing (e.g., computer vision algorithms/Deep Learning)
  - High flexibility (more than FPGAs less than micro-controllers)
  - High-performance (comparable to FPGAs for some applications)
  - Poor energy-efficiency  due to large hardware
  - Development time: often comparable to CPUs
  - $C\_NRE$ low, $C\_RE$ high (comparable or a bit lower than FPGAs)
  - Radiation tolerance:  bad (it can be improved in rad-hard GPUs)

  - GPUs might be interesting to enable High Performance Computing of different apps, including training

# Commercial AI Chips

- **Application specific circuits designed to optimize AI inference:**
  - Limited flexibility (High flexibility but limited to AI algorithms)
  - High energy-efficiency & Performance (trade-offs comparable to FPGAs)
    - Hardware optimized for AI inference
  - Low to medium development time: depending on the development flow
    - $C_{NRE}$ : low/average
    - $C_{RE}$ : low/average
    - Radiation tolerance: low/average (commercial devices)

- Typically used for AI inference on nano/micro-satellites

# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

# Technical Challenges

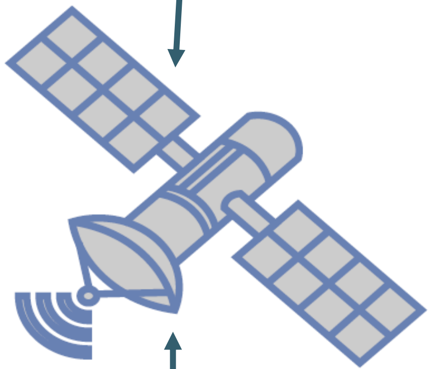
## Communication constraints

- Limited communication bandwidth
- Limited line of sight

## Power Constraints

- Limited power budget
- Depends on orbit & satellite design

## Mission-specific Constraints



## Thermal Management

- Restrict the time for processing
- Depends on orbit & satellite design

## Radiation Effects

- Damages and systems restart
- Corrupted data

## Limited dataset availability

- New onboard sensory
- Not representative datasets (e.g., raw data, domain gap)

# Limited dataset representativeness: domain

- **Domain gap problem**
  - AI models are trained on a *source dataset*
  - In their operational scenario (i.e., on board) AI models work on a *target dataset* featuring a “domain shift” compared to the source dataset
- **What can cause a domain shift in EO?**
  - Different operational conditions (e.g., acquisition time)

Credits: Rens Van Der Linden

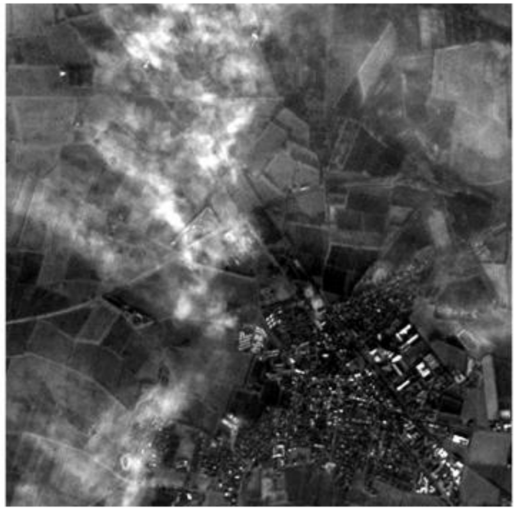
# Limited dataset representativeness: domain

- **What can cause a domain shift in EO?**
  - Different operational conditions (e.g., acquisition time)
  - Not sufficient comprehensive training set (e.g., containing limited geographical areas, limited samples in a time series)

# Limited dataset representativeness: domain

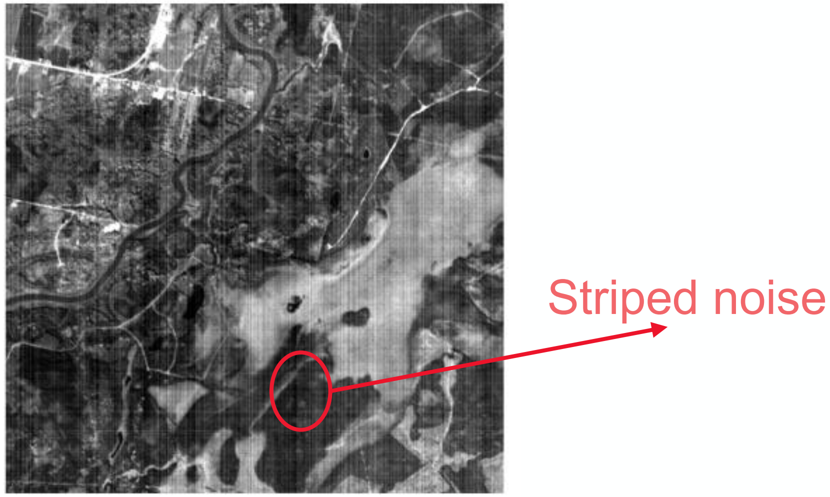
- **What can cause a domain shift in EO?**
  - Unavailability of target data at training time (for instance, when the target sensor is new)
  - Sensor degradation and noise drifting

**Source domain**



Simulated Φsat-2 acquisition – L1 level

**Target domain**



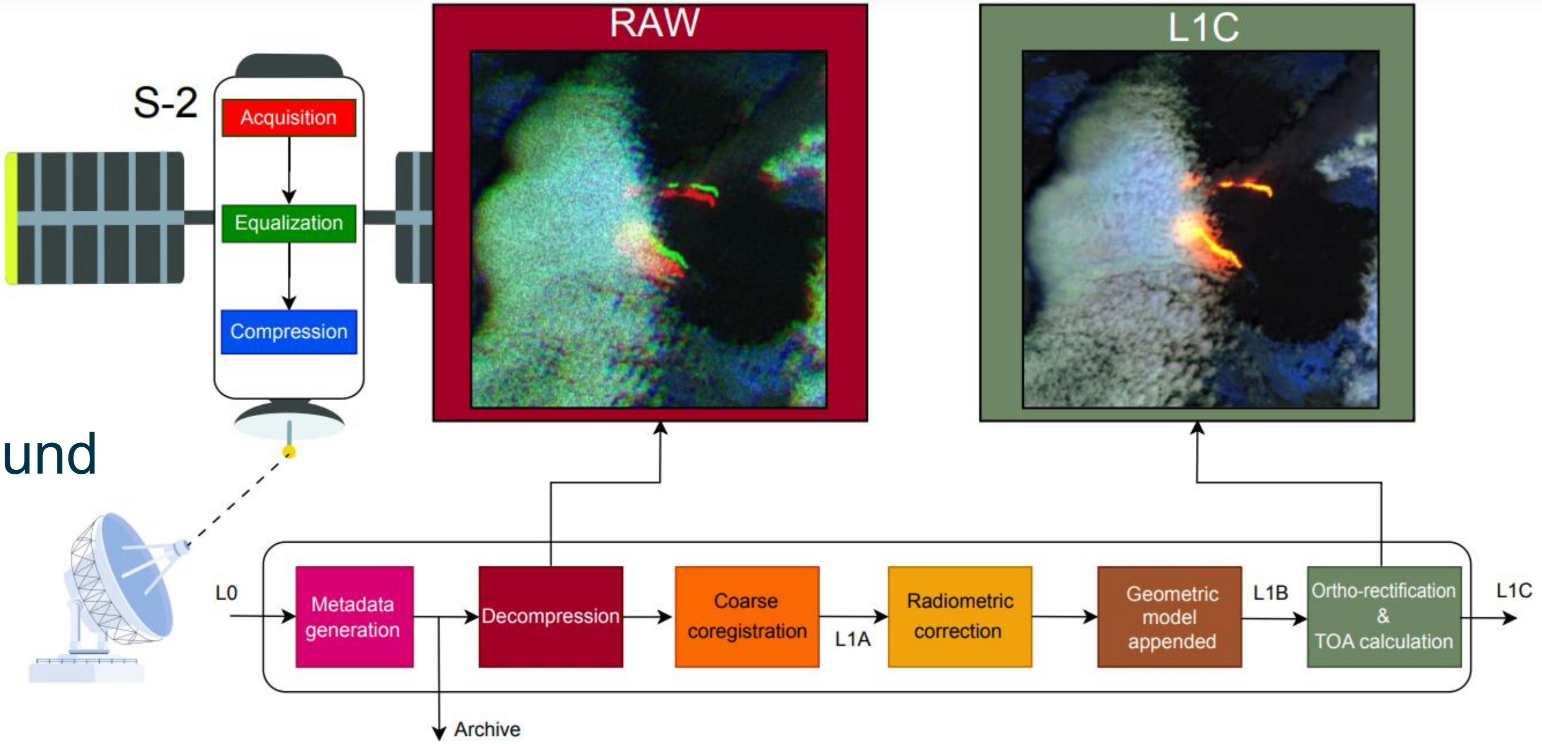

Real Φsat-2 Acquisition – L1 level

Credit: Giorgia Guerrisi

# Limited dataset representativeness: domain

- **What can cause a domain shift in EO?**
  - Lack of fidelity in reproducing onboard data processing chains

- Onboard AI demonstrators are typically made using high-end processing levels (e.g., L1C)
- However, reproducing the on-ground processing chains on board is typically unfeasible



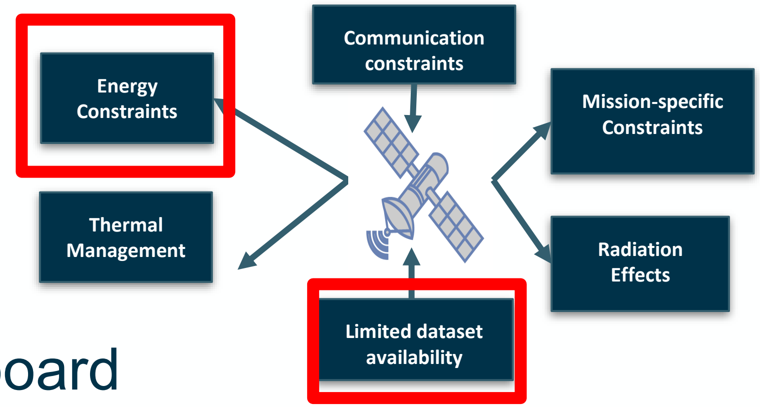

Reproducing onboard processing chains require raw data, which are often unavailable, even for sensors for which higher level products are publicly released (e.g., Sentinel-2)

# Domain gap mitigation strategies

1. **Data download, fine-tuning, and models retraining**
   - Currently adopted in most of EO missions leveraging new sensors
   - Simple approach but leads to longer in-orbit commissioning or operational missions overhead and poor automation
2. **In-orbit domain-gap adaptation techniques**
   - Techniques currently experimented on satellite imagery aim at performing continual domain gap assessment and adaptation via model fine-tuning
   - Might require additional hardware (e.g., GPUs) to speed up computation
3. **Using Foundation Models (FMs) on board**
   - Being pre-trained on massive amount of data, FMs could be more resilient to domain shifts
   - FMs are computationally expensive. Onboard deployment is tricky.

# Energy constraints

- Energy availability fixes the maximum area that can be acquired and processed on board within an orbital period
- **Image pre-processing**
  - Data used on the ground are high-end post-processed
  - Same level of post-processing is often not achievable on board due to power limitations
  - Pre-processing creates significant latency (even higher than AI inference!) that is critical for some applications (e.g., natural disasters)
- **Idea**: Can AI process directly raw data (with minimal pre-processing)?
- **Case study**: thermal anomaly (i.e., volcanic eruption/wildfire) patch classification on Sentinel-2 raw data



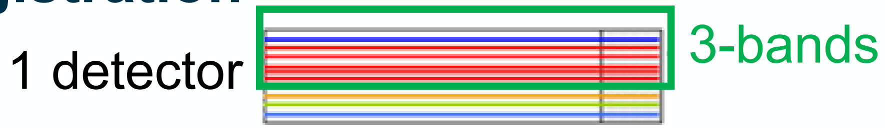
# How Sentinel-2 raw data look like (1/2)?

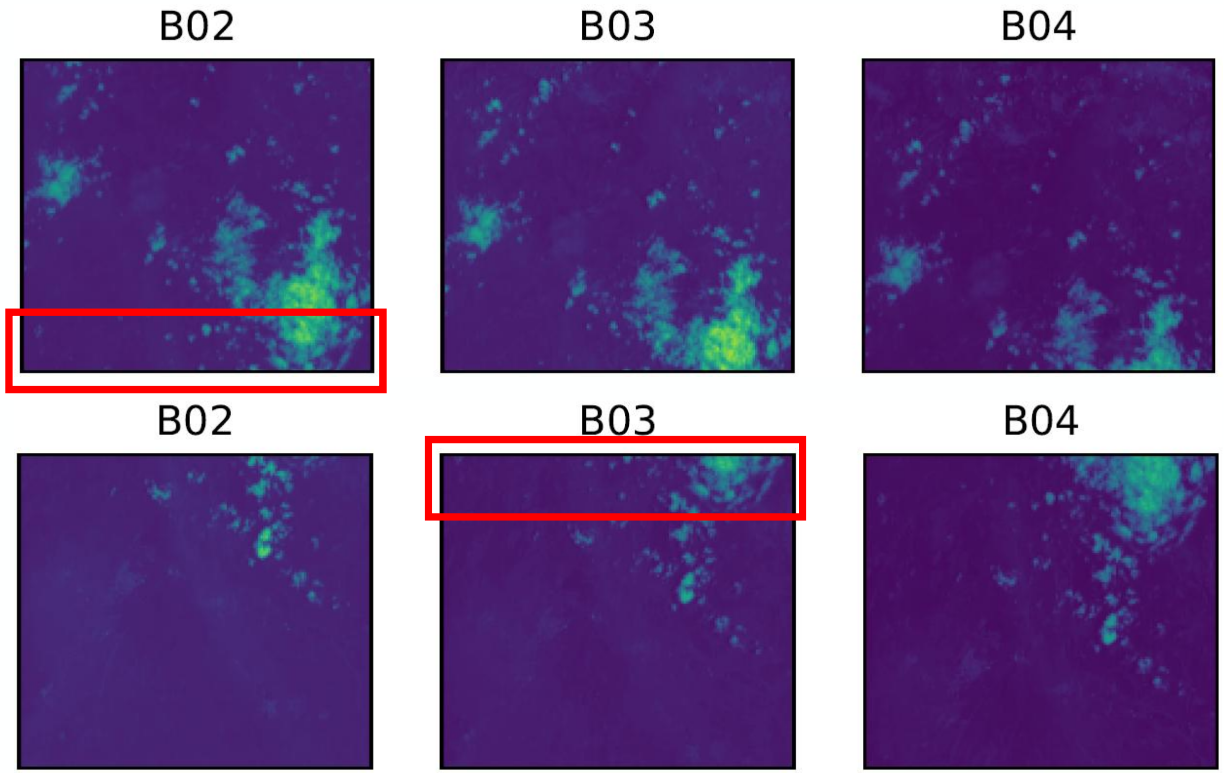
- Sentinel-2 is a **multispectral pushbroom sensor:**
  - **Multispectral – 13** bands in the Visible, Near-Infrared, SWIR ranges
  - **Pushbroom**:
    - Satellite acquire by «sweeping» over a certain area along track
    - An acquisition stops after 3.6 s.
    - The area acquired of a single detector in 3.6 s is called **Granule**
    - Parallaxis between bands in a detector make bands acquire different area

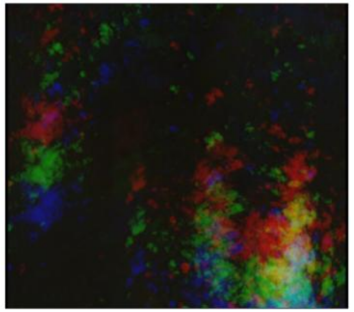
      (**Bands are not registered**)

# How Sentinel-2 raw data look like (2/2)?

- **Lack of bands registration**



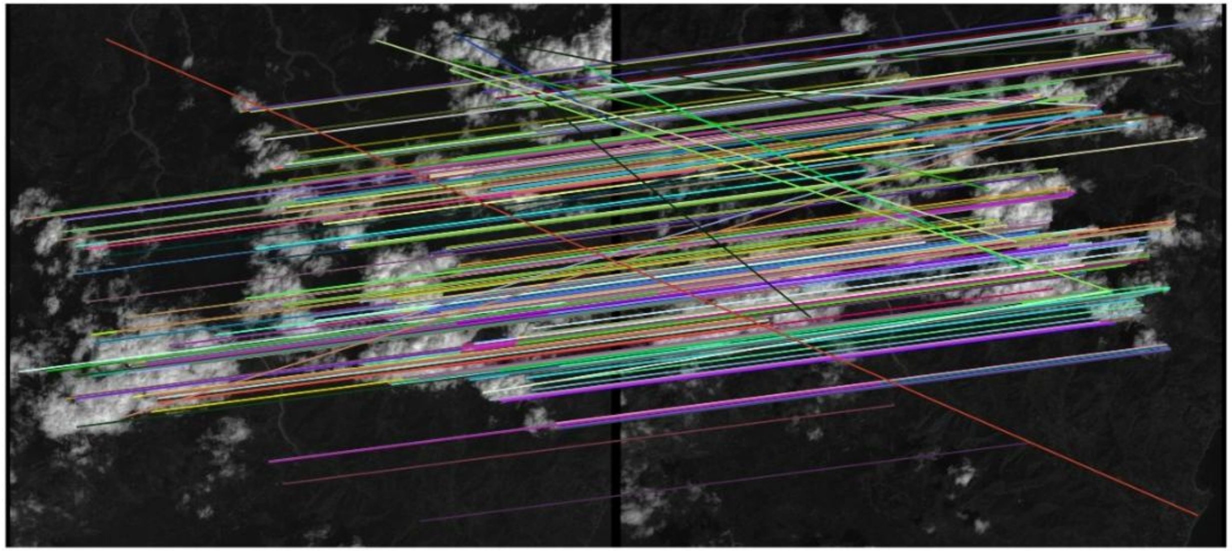



- In addition, **no radiometric processing** (except for bans equalisation)

# Classical bands co-registration techniques





- There are several coregistration techniques (based on computer vision or mechanical model)
- **Computer vision-based bands displacement correction**
  - Most of them use computer vision techniques to extract and match keypoints
  - The average key-points distance provide the registration shift to be matches
- In addition to displacement, some registration corrects errors in rotation, warping, etc.

# Classical bands co-registration techniques

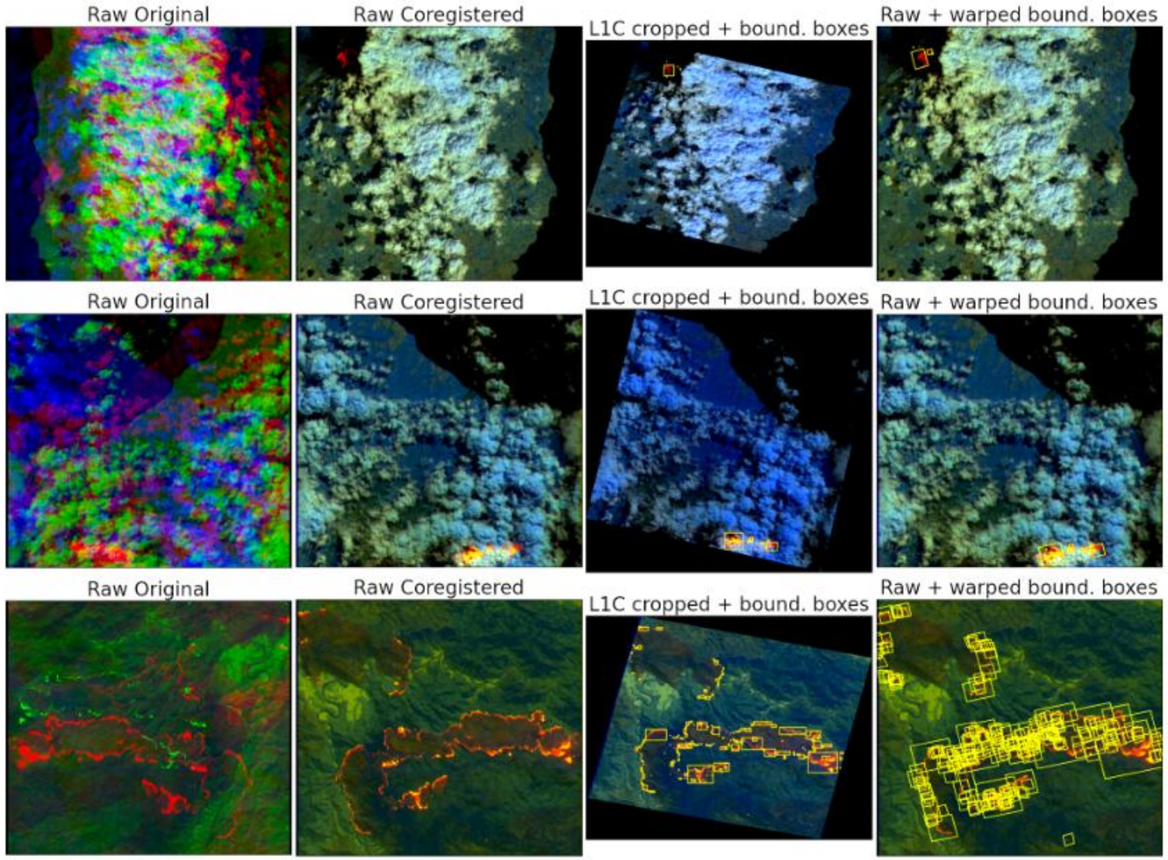
- **THRaWS** (Thermal Hotspots in Raw Sentinel-2 data) dataset:
  - 135 granules containing volcanic eruptions and wildfires
  - 11 granules with no thermal hotspots
  - Bands B8A, B11, B12

- **PyRawS** (Python for Raw Sentinel-2 data)

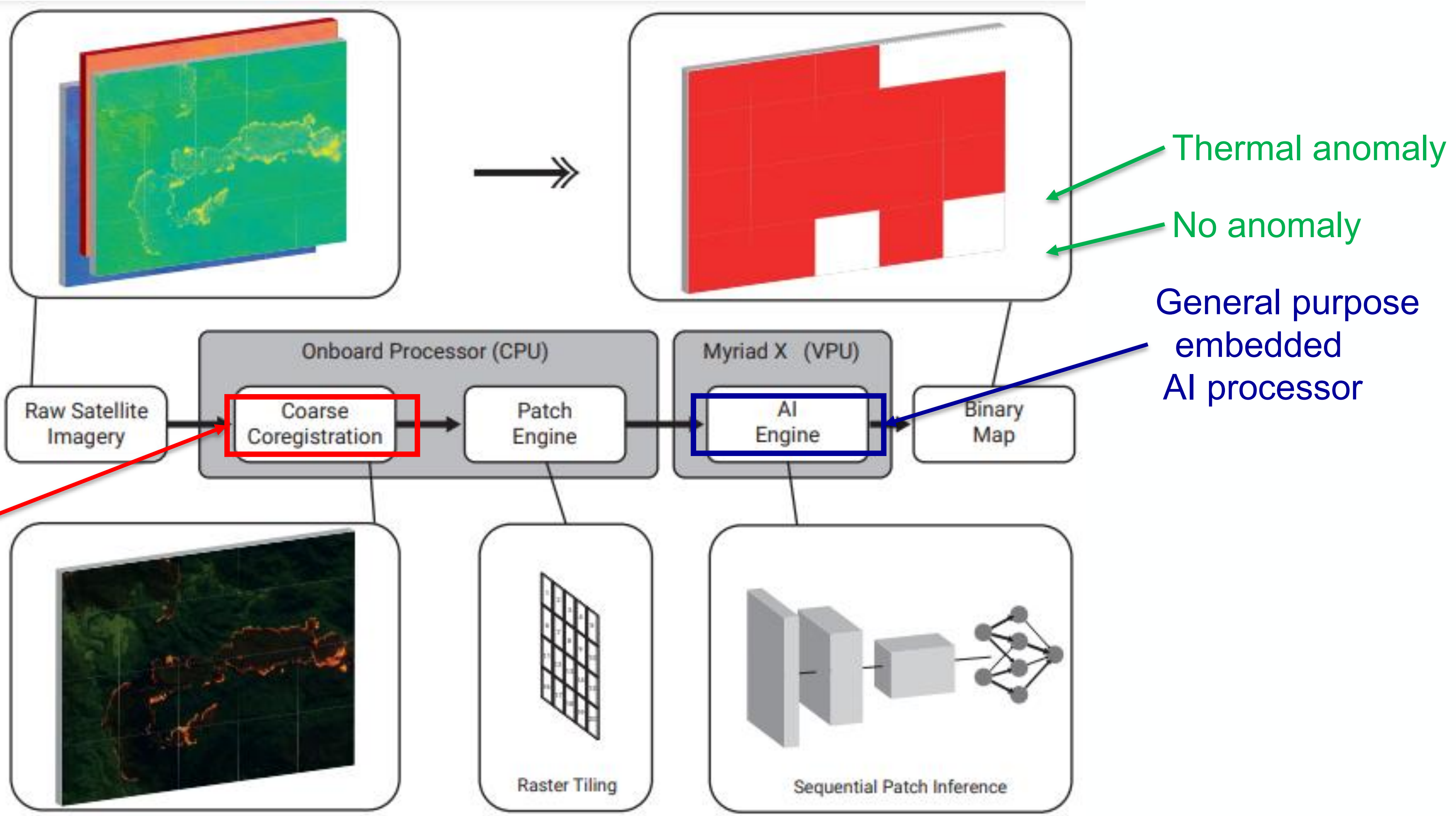
PyRaws

(https://github.com/ESA-PhiLab/PyRawS)

Download THRawS

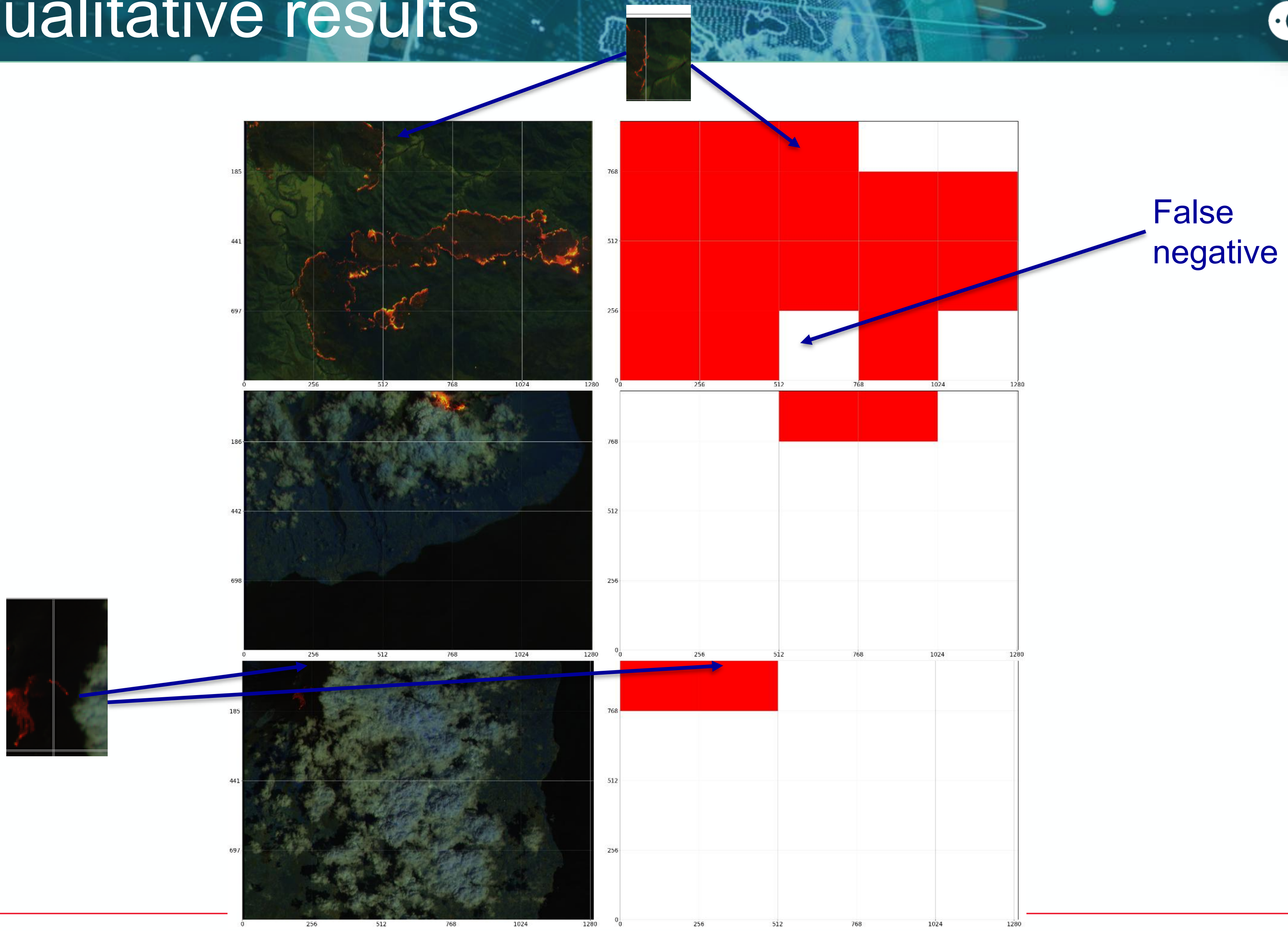


# Onboard END2END processing techniques




Applying fixed precalculated shift (not perfect)

# Some qualitative results



# Energy results

- **Testing set-up**:
  - Onboard processor: Raspberry PI 4 (4 GB)
  - AI board: CogniSat-XE1 board:
    - Flight-heritage: flying on Ubotica CogniSat-6

- **Results**:
  - Latency: 1.8 s < 3.6 s (Real time processing)
  - Power consumption:
    - Average: 5.5 W
    - Peak: 6.4 W

In collaboration with Ubotica Technologies,TU Delft, and University Federico II

# Outline

- Edge computing & edge computing in space
- Applications of Artificial Intelligence in space
- A brief introduction to hardware for payload data processing
- Technical challenges/research trends
- Conclusions

# Conclusion

- Edge computing for space is attracting the interest of research community
- Numerous technical challenges are still existing

- Further research and technology development is needed to make systems based on edge computing in space operational

# Working with us is very easy

- Visiting Professor
- Visiting Researchers (Industrial and Scientific)
- ESA Research Fellowships
- ESA Co-funded PhD
- ESA Early Graduate Traineeships (EGT), Internships, National trainee

# Join ESA Φ-lab through CIN

The **Collaborative Innovation Network (CIN) by ESA Φ-lab**, provides to leading researchers and University Professors the **opportunity to** join ESA Φ-lab and be actively involved in **accelerating the future of Earth Observation with ESA**.

The ESA Φ-lab CIN aims to:

Establish a global network through which researchers and innovators can **JOIN ESA Φ-LAB**
**Promote knowledge** sharing and develop groundbreaking EO solutions

**Check the open calls**

**Follow CIN on Linkedin**

# What to go deeper into Φ-lab disruptive innovation?

Don’t miss our spotlight session at Big Data from Space
(Day 2 – October 2nd at 17:50)