APPLICATIONS AND ENABLING TECHNOLOGIES FOR ON-BOARD PROCESSING AND INFORMATION EXTRACTION: TRENDS AND NEEDS

Murray Ireland[1], Craig Hay[1], Doug McNeil[1], Derek O'Callaghan[2], Sheila McBreen[2], and Roberto Camarero[3]

[1]*Craft Prospect Ltd, Glasgow, Scotland*
[2]*School of Physics, University College Dublin, Dublin, Ireland*
[3]*European Space Agency, Noordwijk, Netherlands*

ABSTRACT

As satellite instruments have increased in capability and complexity, the data generated by them in-orbit has grown to the stage where typical downlink bandwidths are inadequate, resulting in data bottlenecks on-board. This leads to valuable and frequently time-critical data being hidden in amongst other data and only discovered when or if they reach the ground. One solution to this bottleneck problem is to increase the available bandwidth by increasing the size of antennae on the satellite and ground, however this has obvious cost implications.

More cost-effective solutions include prioritising the high-value and time-critical data ahead of other items and simply reducing the total amount of data to be downlinked, such that the bottleneck is eliminated. The former solution relies on the ability to discern value in the data prior to downlink, while the latter requires that data can be reduced (either the number of items or the size of a single item) without loss of valuable information. These activities are examples of information extraction and data processing, where information or value in raw data is inferred on-board, and then used to drive content-sensitive processing of the data to improve data throughput, latency, timeliness and comprehension, with consequential benefits to mission cost and data management.

In this paper, use cases for Earth observation data are summarised. From these, generic on-board applications are defined which deliver data to meet these use cases and provide a number of benefits over traditional data provision approaches. These applications are enabled by recent advances in data processing methods, algorithms and training datasets, which are described. Finally, the requirements of a processing architecture to support these application on-board is presented, derived from an understanding of the system requirements and implications of adopting such data autonomy on-board.

Key words: processing, AI, machine learning, information, requirements, algorithm, dataset.

1. INTRODUCTION

The growing interest in on-board data processing for Earth observation satellites has been driven by both the needs of ground-based applications of satellite data and the increasing challenges borne by the latest on-board instrument technologies. On the application side, end users of satellite data are seeking methods of acquiring more timely data and lower latencies, supplied in a form that is immediately useful to them. On the technology side, instrument manufacturers are developing payloads with higher spatial and spectral resolutions and greatly increased acquisition rates. Such rates far exceed typical current and near-term downlink bandwidths, resulting in severe data bottlenecks.

Addressing the application needs leads to solutions which also address the challenges of adopting these new instruments. On-board processing activities which can extract information from data, reduce the data and prioritise useful information can not only deliver data to end users faster and in a more useful form, they can also filter and intelligently compress data to reduce or eliminate the downlink bottleneck. These on-board activities are many and variable, depending on the on-board functionality desired and the end user requirements to meet.

This work was completed as part of the ESA Technology Research Programme activity "Future Onboard Processing and Information Extraction Algorithms" (FOPIEA). The goal of this work was to identify suitable on-board processing applications and technical solutions (algorithm, dataset and hardware) with which to implement the applications, leading to two breadboard demonstrators.

In this paper, generic on-board applications are presented which may be configured to target groupings of use cases and provide measurable end benefits to mission stakeholders. These use cases have been solicited from a comprehensive survey, a group workshop and one-to-one engagements with remote sensing end users. These applications are underpinned by enabling technologies which leverage the state-of-the-art in embedded processing and AI. These technologies include machine learning algorithms (both traditional and deep learning), training datasets, processing architectures and embedded computing devices. The paper summarises the algorithms which can be used to implement the on-board applications and the factors affecting datasets which can be used to train and test these algorithms. Finally, a summary of the requirements and further implications of adopting these technologies to implement the proposed applications is presented. This leads to several impact areas and corresponding needs that must be addressed in the design of future on-board processing systems.

2. USE CASES AND APPLICATIONS

Remote sensing use cases are the driver of any satellite EO activities, whether a traditional mission such as the Copernicus Sentinels or one of the many NewSpace missions leveraging some degree of on-board processing. While the technical feasibility of applications is one concern, and partially addressed in Section 3, the usefulness of any application is dependent on the need for it, and this is driven by the many remote sensing use cases that exist.

2.1. Use Cases

The following use cases have been identified through literary sources, one-to-one engagements with remote sensing specialists and group workshops. Remote sensing use cases are many, with end users spread across industry, academia and government. While there can be overlap in many use cases, they can broadly be grouped into the following categories.

Agriculture EO data enables farmers, landowners and other stakeholders to map, monitor and manage land, crops and other related factors, with the goal of making their operations more efficient, reducing costs and maximising yield [1]. Data can also be used to monitor, predict and manage the risk of natural and manmade disasters or damage. On a larger scale, EO data can be used by governments and other bodies to detect and monitor for illicit farming.

Defence Defence and security practices often leverage capabilities in sensing, monitoring and tracking objects or targets of interest [1, 2]. While also including ground and aerial surveillance and reconnaissance, such activities have made use of satellite-based observations for some time now. Typical defence and security use cases include mapping of areas of interest, detection and identification of vehicles and infrastructure and monitoring of specific sites of interest. Given the sensitive nature of such activities, any insights extracted from satellite data demand a high confidence in their accuracy. Specific activities will have additional requirements in the timeliness, quality and persistence of data.

Disaster Response EO data can enable the detection, monitoring and analysis of natural and manmade disasters, and facilitate timely responses to the disasters. A quick response to events such as severe storms and earthquakes can significantly reduce loss of life, while some disasters such as wildfires and floods can be mitigated or arrested entirely by sufficiently early warning. Additionally, the conditions in which disasters arise can be identified in data of sufficient accuracy and quality, including those in which disasters are likely to occur [1, 3, 4].

Forestry and Vegetation Multispectral satellite imagery is a common enabler for forestry and vegetation monitoring due to the benefit of infrared data in visualising vegetation. In this context, EO data can be used to perform forest-specific land cover classification, perform change detection to monitor coverage and health and estimate damage due to deforestation, fires, floods, storms and other disasters [5].

Hydrology Hydrological use of EO data involves the measuring and monitoring of coastlines, inland waterways and movement of ice, including flow and icebergs [1, 5]. Just as land use can be monitored, so can use of coastal and inland water regions. More advanced information, such as water levels, can be obtained using a combination of optical data and measurements from other instruments such as altimeters [6].

Land Use and Land Cover Land use and land cover mapping is commonly augmented by satellite data [7]. In addition to providing information to create online and physical maps, this information can also be used to monitor urban growth, enable infrastructure planning and perform large-scale change detection on urban and other regions.

Meteorology Meteorological applications of EO data include weather monitoring and prediction and atmospheric science [8, 9]. Some aspects of meteorology fall under disaster response, such as the identification of severe storms, requiring quick response as with other natural disasters. The majority of use cases involve creation and informing models of weather systems, with the goal of providing accurate weather predictions and creating greater understanding to benefit other areas such as climate change and resilience.

Maritime Maritime applications of EO data are primarily focussed on ship detection and tracking [1, 10]. Use cases include identifying illegal activities, such as ships without AIS (automatic identification system) or those involved in oil spills and environmental law violation, detecting vessels in distress and monitoring traffic in ports and other high traffic regions. Additional use cases include monitoring of fisheries and objects or events at sea, such iceberg detection and oil spill monitoring.

Resource Management Natural resource management makes use of both land cover mapping, for classifying and categorising areas with potential for extraction of natural resources, and change detection, for monitoring existing infrastructure such as mines, oil rigs and renewable energy sites [1, 11].

2.2. Applications

The above use cases vary in terms of the instruments they employ to capture the data, the processing applied to this data and the ultimate form of and insights that are extracted from the data. However, they may be generalised in terms of the operational concept used to implement each use case. For example, many hydrological use cases involve persistent monitoring of a specific location or region. This activity also applies to agricultural and land asset monitoring. In each use case, the end user and data format can be very different, but the operational concept and data processing pipeline are broadly similar.

Nine generic applications have therefore been defined. These applications each relate to a single operational concept, defining the region of interest, duty cycles, revisit requirements, etc. They can then be configured for a specific use case (e.g. disaster monitoring, ship tracking) by selecting appropriate enabling solutions, algorithms and training datasets.

Event Detection An event that has occurred or is occurring is automatically detected and a notification or alert automatically queued for downlink on the next ground station pass. The notification can include simply the type of event that is occurring, where it is occurring and when it was identified, or it can be a larger report, indicating the severity and risk (for disasters) or scale (e.g. for atmospheric events). Use cases include real-time detection of disasters, illegal activities and urban events such as demolitions or protests.

Event Prediction The conditions for the occurrence of specific terrestrial or atmospheric events, typically natural or manmade disasters or severe weather phenomena, are predicted and identified onboard the satellite. The satellite notifies end users such of the probability of the event occurring and additional information such as location and timeframe. Prediction of events based on instrument data alone is challenging, and so data fusion would be required to employ secondary sources such as additional instruments, on-board sensors and periodically-updated databases. Use cases are typically limited to naturally-occurring events such as weather phenomena and natural disasters, although it may be possible to predict some unintentional human-caused events such as accidents or oil spills.

Event Monitoring Event monitoring focusses on high-frequency reporting of a target following an event such as a natural or manmade disaster. It differs from event detection in that the event and its location is presumably already known and must be monitored for some period of time. The changes to a location or feature following an event are often, but not exclusively, transient and on a shorter timescale than typical EO changes. The satellite returns information on the event in an ongoing capacity, reporting on absolute properties (e.g. scientific measurements, severity, etc.) and changes (e.g. loss of forest due to wildfire, spread of oil, etc.). Use cases include monitoring of disasters, illicit activities and more innocuous events such as protests.

Persistent Monitoring Monitoring of permanent or less-transient features constitutes a distinct monitoring application from monitoring of a specific event. In this case, a single or multiple locations or assets are monitored and reported on a regular (or change-triggered) frequency over a long period of time. The satellite returns a regular report on the target, which may be delivered on a frequency matching that predicted of changes in the target, or this report could be specifically triggered when changes are sufficiently distinct. Use cases include monitoring of agriculture, environment in various sensitive locations (such as coastlines) and work sites for surveillance of ground assets.

Damage Assessment Locations, structures or other targets which have suffered damage are targeted for assessment. Causes of damage may be included in the assessment, but severity and information to enable response or recovery are more critical. Use cases include assessment of building damage, damage to agriculture due to flooding, drought or otherwise and cause-specific instances such as damage due to wildfire, flooding or storms.

Global Mapping The entirety or majority of the planet is imaged to provide a global map of some set of features, typically land cover (water, forestry, urban, etc.) or land use (fishing, recreation, industrial, etc.). Coverage may be truly global, or focussed on a specific terrain type, e.g. urban, water, rainforest. Land cover is readily identifiable in satellite imagery. For other use cases, sufficient input data quality or data fusion is required to extract the desired information.

Localised Mapping A set of features are imaged and mapped for specific location(s). This could be standard land use mapping targeted at a specific region (e.g. country, state) or may be more precise, returning mask layers indicating properties such as crop health or exploitation potential. In the latter case, sufficient input data quality or data fusion is required to extract the desired information. This application differs from persistent monitoring in that it is concerned with identifying the content of a location via raster/vector maps without reporting any more detailed analysis.

Object Detection & Tracking A specific or set of objects/targets are detected from orbit, reported to end users, and tracked to enable an appropriate ground response, e.g. law enforcement. Targets may be specific to an environment, such as security violations at specific sites, or coverage may be global, such as illegal ships ocean-wide. Use cases are typically defence and security-related, but can also include scientific applications such as animal tracking for migration research, etc.

Scientific Measurement A specific property is measured from orbit, typically using non-optical data such as SAR, microwave radiometry, altimetry and LiDAR. This property can be provided as a simple measurement (e.g.

moisture level for specific field) or as a single-channel, georeferenced “heatmap” (or equivalent for non-image data), where pixel intensities related to measurement values. Use cases include biomass measurement, elevation mapping, global temperature mapping and soil moisture level measurement, among others.

3. INFORMATION EXTRACTION ALGORITHMS

The applications described in Section 2.2 can be realised by implementing various algorithms to support pre-processing, information extraction and data autonomy tasks. Information extraction tasks are typically performed by machine learning models. These can be traditional, such as random forest or support vector machines, or more modern solutions such as deep learning models. This section briefly summarises the available algorithms for information extraction, focussing on these with prior application to remote sensing.

3.1. Traditional Methods

There are several classical machine learning methods which are popular for classification tasks in particular, with applicability to other tasks such as regression and change detection.

Pre-Processing and Feature Extraction When using classical machine learning algorithms, feature extraction is an important first stage of the pipeline in many applications. Unlike deep learning, where it is common for the model to learn from large amounts of training data which features of the data are important, classical machine learning algorithms generally perform better when the number of features in the data has been reduced.

Common methods of pre-processing and feature extraction include Principal Component Analysis (PCA), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), and factor analysis.

Support Vector Machines A support vector machine (SVM) [12] is a supervised learning model, linearly separating multiple classes in a high dimensional space. SVMs are widely used in remote sensing data processing pipelines as the final classification step [13]. They have previously been implemented on-board the EO-1 spacecraft, processing data from the Hyperion instrument [14].

Random Forest Random forest is an ensemble learning method [15], combining multiple models derived from other algorithms and returning a summary statistic of the prediction or classification produced by each model. In this way, overfitting is avoided.

Naïve Bayes Naïve Bayes classifiers are simple classifiers that rely on Bayes’ theorem. They are “naïve” in the sense that independence between each element is assumed.

Strengths of Naïve Bayes classifiers include their simplicity and ease of implementation, good performance and scalability. One weakness is that the assumption of conditional independence is rarely true, another it is that they are often surpassed in performance by other types of models when properly trained and tuned.

K-Nearest Neighbour K-Nearest Neighbour (k-NN) algorithm is a supervised algorithm that can be used for both classification and regression tasks. It works on the assumption that samples that belong to the same class or have the same numerical properties (in regression applications) are in close proximity in the feature space [16, 17, 18].

3.2. Deep Learning Methods

Deep learning solutions can be categorised by the architecture type, although some specific architectures are named where these are prominent. Benefits of deep learning methods over traditional methods are primarily in their versatility, lacking the more rigid structure required by traditional methods.

Convolutional Neural Networks Convolutional neural networks (CNN) are deep learning models which accept 2D or 3D input arrays (typically images) and return outputs relating to the presence of features of interest within the array. They do so by performing convolutions and other mathematical operations on the input data across several layers. A CNN can be trained using supervised learning, where the features of interest are known and specified, or unsupervised learning, where the features are unknown.

CNNs can be used to solve classification problems and can also be used for other tasks such as change detection and anomaly detection.

Object Detection Networks Object detection networks are a subset of CNN where the returned result includes both identified features and a location within the data, typically described by a bounding box. Popular implementations include Single Shot Detector and You Only Look Once (YOLO) [19].

Semantic Segmentation Networks Semantic segmentation networks are a subset of CNN where the classification is performed on each pixel in the data. Of semantic segmentation networks, U-Net is one of the most prominent [20], with some effort made to reduce its size for deployment on embedded devices [21].

Autoencoders Autoencoders are deep neural networks that reproduce their input at their output layer. With the correct training methods, such networks are more effective that traditional compression techniques such as PCA [22] and can be used to learn an efficient representation for dimensionality reduction. They can also be used for error correction [23].

Neural Regression Neural regression models return values from a continuous number set rather than discrete outputs, as is the case with classification.

3.3. Datasets

Key to the successful use of any machine learning model is training data. With the ever-growing use of machine learning in remote sensing sciences, labelled datasets derived from a variety of EO missions have appeared, targeting a variety of applications. Of these datasets, there are a variety of factors that determine their suitability for a given use case and on-board processing application. These factors are summarised in this section.

Some popular datasets include:

BigEarthNet A land cover dataset for classification, derived from Sentinel-1 and 2 [24].

DOTA A series of object detection datasets with a large number of labels, derived from Google Earth and satellite and drone imagery [25].

EuroSAT A land cover dataset for classification, derived from Sentinel-2 [26].

SEN12MS A large land cover dataset for semantic segmentation, derived from Sentinel-1 and 2 and using MODIS land cover rasters as labels [27].

SpaceNet A series of datasets for several different applications [28].

xView Two datasets related to object detection and building damage, respectively [29, 30].

The suitability factors of these and other datasets are described below.

Application Typically indicated by the labels or annotations used in the dataset. For example, land cover datasets can include generic labels such as "agriculture", "urban", and "forest" or use those derived from a particular classification such as CORINE or MODIS. Labels for classification tasks may describe specific features (e.g. "ship", "cloud", "burnt area") or discrete states such as damage or cloud cover levels. Labels for regression tasks can be continuous, for example describing scientific properties such as height or moisture level.

The dataset labels have the greatest impact on the use cases that can be targeted with any network trained on the dataset.

Label Format Labels may be specified as a single string describing a training chip or tile, a string with bounding box coordinates or a binary pixel mask with an associate annotation. The format provided impacts the classification tasks that can be performed. Labelled chips and tiles are suitable for simple chip classification tasks, labelled bounding boxes are best-suited for object detection and pixel masks are intended for semantic segmentation tasks.

Ground Truth The veracity of a dataset's labels is important for assurance of any IE tasks trained on the dataset. Creation of datasets by fusing georeferenced images with another data source (e.g. weather phenomena, land cover databases, emergency alerts) is relatively straightforward. More difficult is ascertaining the accuracy of these labels.

Uncorrected, poor ground truth can result in difficulty training the network and inaccurate inference results.

Size and Diversity As with all machine learning problems, a sufficient quantity and diversity of training samples are required to ensure the network is both accurate and sufficiently able to generalise. The diversity of the samples should meet the requirements of the use case and mission concept, reflecting variations in geographical location, time of day or year and in the appearance of features.

Instrument Source A key issue for on-board machine learning. While many ground-based ML use cases use similar sources (e.g. Landsat 8, Sentinel-1, Sentinel-2) for both dataset training and analysis, this is often not the case with on-board applications. Many datasets are derived from the Sentinel-2 MSI or the Landsat 8 OLI, and the instrument ultimately providing data for processing in-orbit is unlikely to match either of these instruments exactly, in terms of the resolution, spectral bands and unique sensor anomalies. Indeed, many on-board models are likely to be trained for instruments which haven't yet been launched, or at haven't at least acquired representative training or test data.

As a consequence, it can be necessary to transform a dataset to better match the input data during inference, or vice-versa. With the additional lack of validation or test data, the true accuracy of the network is unknown until in-orbit testing, at which point training updates will almost certainly be required.

Processing Level As with the previous factor, the processing or product level of the data is also a key issue for on-board ML. The majority of datasets employ data which has undergone some sort of pre-processing, such as radiometric and geometric correction and georeferencing. As a result, the instrument data must undergo the same pre-processing during operation so that the model recognises it. Alternatively, a model trained on raw data may be able to recognise data directly from the instrument. There are many challenges to this approach, however, including lack of raw training datasets, anomalies in raw data and the format itself of raw data.

Licence Many labelled datasets are open access and available for both research and commercial use. Others, typically those derived from commercial missions and platforms such as Google Earth Engine, are far more restrictive. Many datasets employ Creative Commons licences, restricting commercial usage or enforcing the sharing of any modifications to the dataset.

4. IMPLICATIONS AND NEEDS OF ON-BOARD PROCESSING APPLICATIONS

Adoption of the described applications and their processing and information extraction tasks impacts the on-board data processing system in a number of ways.

Interfaces Interfaces with data handling and data storage components are required to handle and store the inputs to and outputs from the processing components, as well as any additional data required, such as neural network weights and reference samples for change detection. In-orbit updates of models and weights also has the additional requirement that both model inputs (raw or processed data) and outputs (information) can be downlinked periodically or at will, to facilitate re-training on the ground.

On-the-Fly vs Offline Processing Due to the acquisition rates of modern instruments and the relatively limited processing hardware currently used on-board EO satellites, real-time processing of instrument data is a challenge. This may be overcome by multi-stage information extraction, for example using a simple, fast model to evaluate cloud cover and pass only data with low cloud cover, storing the remainder for offline processing during eclipse.

Reconfigurability Reconfiguration of algorithms such as ML models and their associated weights is a strict requirement for the processing system, the command and data handling (C&DH) subsystem and the uplink infrastructure. Support for file uplinks which take multiple passes is vital to enabling ongoing improvement of ML models and other algorithms. Without this capability, the satellite would be limited to use of algorithms as packaged at launch. This may be acceptable where the exact flight configuration used for processing has been flown before and test data is available, but not otherwise. More capable uplinks, featuring rates from tens to hundreds of Mbps, will also be very desirable to support these file uploads. A combination of both multi-pass uplinks and increased bandwidth is recommended to provide sufficient future proofing.

Payload reconfiguration is also impacted by on-board processing where autonomous quality checks or anomaly detection can lead to payload performance improvements such as cal/val and active optics adjustments.

Fault Tolerance Fault tolerance must be considered on various levels of the system hierarchy and in both software and hardware. Many space-grade devices include built-in error checking, while component redundancy is common in newer devices with less space heritage. In devices without built-in error checking, it is necessary to consider the impact of deploying a computationally-intensive and data-critical algorithms such as neural nets. A portion of the FPGA logic, for example, must be reserved for error checking and the size of the network or other algorithms as deployed on the device must facilitate this.

Criticality Data processing tasks are typically considered non-mission-critical in that they are unlikely to affect critical systems such as power, ADCS and comms. A move towards mission autonomy tasks (potentially driven by outputs of the data processing system) will increase the criticality of any autonomous components. The large state space of any ML components makes assessment of the reliability and replicability these components and the large system a significant issue which must be carefully addressed during development time and during operation with sufficient error checking and fault mitigation strategies.

5. REFERENCE PROCESSING ARCHITECTURE

Figure 1 shows a functional architecture for on-board data processing and management. This architecture has been derived to meet the needs of a generic EO application. The operation of the processing system can be described as follows:

1. The payload data is acquired by the payload in its raw format.
2. The raw payload data is pre-processed to correct for radiometric and geometric defects and format the data such that it is recognisable and comprehensible by the information extraction algorithms.
3. The processed data is passed through one or more information extraction algorithms to extract information such as features, masks, value and other metadata.
4. The extracted information is used to create data products comprising metadata and the data in a specified form, e.g. georeferenced tile, feature mask, etc. It may also be reduced at this point by compressing or removing low-value regions of the data. Data value can be determined at this point and included in the product.
5. The compressed data products are then indexed and ordered for downlink prioritisation based on their perceived value.
6. The queued data is downlinked on the next ground station pass.

The architecture is annotated for the following:

(a) Raw data acquired by the payload is stored in mass memory immediately following acquisition. This ensures that the original data is preserved in the event of an anomaly in the processing pipeline and can be compressed and downlink as is currently done on institutional missions.

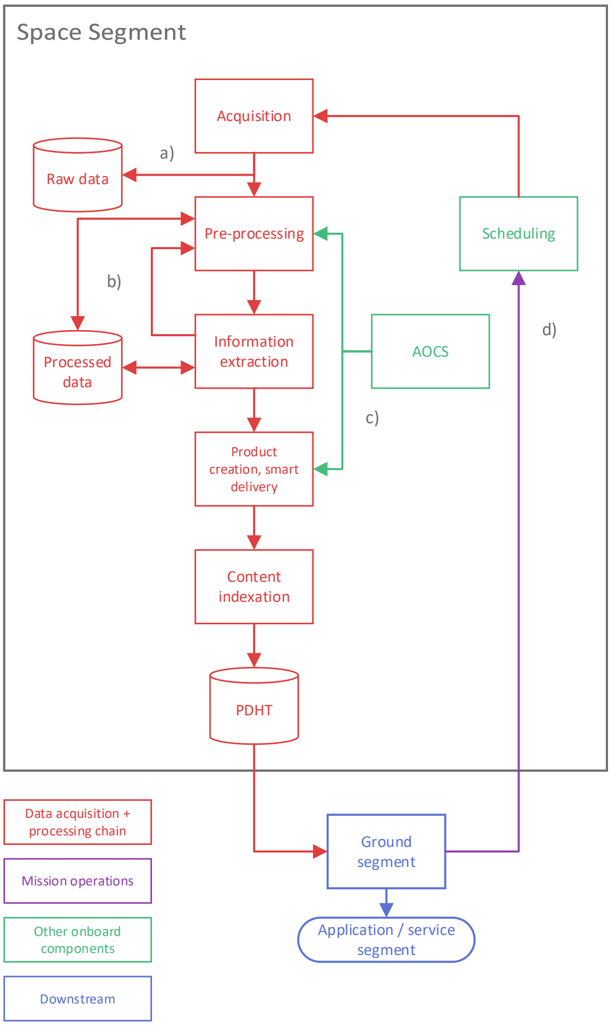

Figure 1. Reference processing architecture for data processing and autonomy.

(b) Further pre-processing of the data may be performed after information extraction. For example, inference with a simple, high-speed algorithm could be performed in real-time to extract some initial insights. Based on these results, the data could then be discarded, stored in mass memory for further processing later or passed back to pre-processing for immediate further processing. For example, anomaly detection may indicate that further geometric corrections are required.

(c) Real-time telemetry may be received from AOCS to enable pre-processing tasks such as radiometric and geometric corrections and data autonomy tasks such as the creation of georeferenced data products and feature geolocations. This is implied use of data fusion.

(d) As data autonomy is limited to processing and decision-making around the payload data, there are no interfaces with mission critical subsystems, including the payload itself. Acquisition activities are performed via onboard schedule or commanding and planned on the ground.

6. CONCLUSIONS

In this paper, the many use cases of Earth observation data have been generalised to allow the definition of generic on-board applications. These on-board applications are comprised of several complementary tasks, enabling EO payload to be acquired, understood and decisions made to improve the throughput, timeliness and value of the data to the benefit of end users. The applications and their constituent tasks are enables by the latest advances in deep learning algorithms and the proliferation of training datasets, suitable for such algorithms and targeting the EO use cases.

A generic processing reference architecture has been defined to meet the needs of these applications. Needs that must be addressed include those of the end user, the requirements of the applications for specific functionality (e.g. classification algorithms, change detection), training datasets for specific use cases and the needs of and other implications on the on-board processing system, such as considerations for error checking, fault tolerance and real-time processing.

The findings described in this paper are currently being referenced in the delivery of two demonstrations of on-board data processing and information extraction, targeting use cases for an institutional and a NewSpace mission. These demonstrators will test the feasibility and benefits of implementing two of the defined applications on-board representative satellite hardware, including space-ready data processing devices.

ACKNOWLEDGEMENTS

The authors would like to thank all participants in the FOPIEA meetings, survey and workshop which informed the direction of this work.

REFERENCES

[1] SSTL. Applications of Earth Observation, 2020.

[2] European Space Imaging. Satellite Imagery for Defence & Intelligence, 2018.

[3] Josie Garthwaite. Mapping dry wildfire fuels with AI and new satellite data, May 2020.

[4] Dave Petley. Detecting landslide precursors from space: the use of Sentinel images, November 2017.

[5] Prashant K. Srivastava, Dawei Han, Miguel Rico Ramirez, and Tanvir Islam. Machine Learning Techniques for Downscaling SMOS Satellite Soil Moisture Using MODIS Land Surface Temperature for Hydrological Application. *Water Resources Management*, 27(8):3127–3144, June 2013.

[6] Dolores Beasley, Marta Metelko, and Lynn Chandler. NASA Satellites Measure and Monitor Sea Level, July 2005. Publisher: Brian Dunbar.

[7] David Holland and Paul Marshall. Updating Maps in a Well-Mapped Country Using High Resolution Satellite Imagery. January 2004.

[8] Tiffany Means. Weather Satellites: Forecasting Earth's Weather From Space, November 2019. Section: ThoughtCo.

[9] John Stafford. Orbital Micro Systems Captures High Resolution Passive Microwave Temperature Soundings of Typhoon Hagibis, October 2019.

[10] Emmanuel Chassot, Sylvain Bonhommeau, Gabriel Reygondeau, Karen Nieto, Jeffrey J. Polovina, Martin Huret, Nicholas K. Dulvy, and Herve Demarcq. Satellite remote sensing for an ecosystem approach to fisheries management. *ICES Journal of Marine Science*, 68(4):651–666, March 2011.

[11] Earth-i. Analyse Natural Resources With Satellite Data, 2021.

[12] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[13] Giorgos Mountrakis, Jungho Im, and Caesar Ogole. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3):247–259, May 2011.

[14] Lukas Mandrake, Umaa Rebbapragada, Kiri L. Wagstaff, David Thompson, Steve Chien, Daniel Tran, Robert T. Pappalardo, Damhnait Gleeson, and Rebecca Castaño. Surface Sulfur Detection via Remote Sensing and Onboard Classification. *ACM Transactions on Intelligent Systems and Technology*, 3(4):1–20, September 2012.

[15] Leo Breiman. Random Forests, January 2001.

[16] Shinya Tanaka, Tomoaki Takahashi, Tomohiro Nishizono, Fumiaki Kitahara, Hideki Saito, Toshiro Iehara, Eiji Kodani, and Yoshio Awaya. Stand Volume Estimation Using the k-NN Technique Combined with Forest Inventory Data, Satellite Image Data and Additional Feature Variables. page 17, 2015.

[17] Yangxiaoyue Liu, Yaping Yang, Wenlong Jing, and Xiafang Yue. Comparison of Different Machine Learning Approaches for Monthly Satellite-Based Soil Moisture Downscaling over Northeast China. page 23, 2018.

[18] Hua Sun, Qing Wang, Guangxing Wang, Hui Lin, Peng Luo, Jiping Li, Siqi Zeng, Xiaoyu Xu, and Lanxiang Ren. Optimizing kNN for Mapping Vegetation Cover of Arid and Semi-Arid Areas Using Landsat images. *Remote Sensing*, 10(8):1248, August 2018.

[19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. *arXiv:1506.02640 [cs]*, May 2016. arXiv: 1506.02640.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597 [cs]*, May 2015. arXiv: 1505.04597.

[21] Nazanin Beheshti and Lennart Johnsson. Squeeze U-Net: A Memory and Energy Efficient Image Segmentation Network. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1495–1504, Seattle, WA, USA, June 2020. IEEE.

[22] Geoffrey E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507, 2006.

[23] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning - ICML '08*, pages 1096–1103, Helsinki, Finland, 2008. ACM Press.

[24] Gencer Sumbul, Marcela Charfuelan, Begum Demir, and Volker Markl. BigEarthNet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 5901–5904, Yokohama, Japan, July 2019. IEEE.

[25] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. DOTA: A Large-scale Dataset for Object Detection in Aerial Images. *arXiv:1711.10398 [cs]*, May 2019. arXiv: 1711.10398.

[26] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *arXiv:1709.00029 [cs]*, February 2019. arXiv: 1709.00029.

[27] M. Schmitt, L. H. Hughes, C. Qiu, and X. X. Zhu. SEN12MS - A Curated Dataset of Georeferenced Multi-Spectral Sentinel-1/2 Imagery for Deep Learning and Data Fusion. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W7:153–160, September 2019.

[28] SpaceNet Partners. SpaceNet Challenge Datasets, 2020.

[29] Darius Lam, Richard Kuzma, Kevin McGee, Samuel Dooley, Michael Laielli, Matthew Klaric, Yaroslav Bulatov, and Brendan McCord. xView: Objects in Context in Overhead Imagery. *arXiv:1802.07856 [cs]*, February 2018. arXiv: 1802.07856.

[30] Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston. xBD: A Dataset for Assessing Building Damage from Satellite Imagery. *arXiv:1911.09296 [cs]*, November 2019. arXiv: 1911.09296.