GABRIELE CAVALLARO, DORA B. HERAS, ZEBIN WU, MANIL MASKEY, SEBASTIÁN LÓPEZ, PIOTR GAWRON, MIHAI COCA, AND MIHAI DATCU

High-Performance and Disruptive Computing in Remote Sensing

HDCRS—A new Working Group of the GRSS Earth Science Informatics Technical Committee

The High-Performance and Disruptive Computing in Remote Sensing (HDCRS) Working Group (WG) was recently established under the IEEE Geoscience and Remote Sensing Society (GRSS) Earth Science Informatics (ESI) Technical Committee to connect a community of interdisciplinary researchers in remote sensing (RS) who specialize in advanced computing technologies, parallel programming models, and scalable algorithms. HDCRS focuses on three major research topics in the context of RS: 1) supercomputing and distributed computing, 2) specialized hardware computing, and 3) quantum computing (QC). This article presents these computing technologies as they play a major role for the development of RS applications. The HDCRS disseminates information and knowledge through educational events and publication activities which will also be introduced in this article.

INTRODUCTION

RS has come a long way since 1858, when Gaspard-Félix Tournachon captured the first aerial photograph from a hot air balloon over the Bièvre Valley in France [1]. At the beginning of 1972, *Landsat* data kickstarted the big data era by capturing images of the whole Earth's surface every two weeks [2]. The development of artificial satellites in the latter half of the 20th century allowed RS to progress to a global scale and monitor the entire planet in high resolution, on demand, and in near-real time.

Since 2008, with the emergence of the free and open data access policy for *Landsat* data [3], [4], many governments and space agencies have opened their archives, making large collections of satellite RS data available to everyone [e.g., the European Space Agency's (ESA's) Copernicus] [5]. RS was and further is a stimulating factor in the development of disruptive and high-performance computing (HPC) technologies. An example is the case of synthetic aperture radar (SAR) image formation. SAR is an active, coherent imaging system operated in the microwave domain. An SAR system records millions of samples per second. The transformation of the received echoes, i.e., the focusing process, requires application of matched filters, principally involving the computation of Fourier transforms. In the early 1960s, this was a major big data and HPC challenge, stimulating the use and development of new technologies. At the time, optical coherent processing was one of the first novelties that HPC technology used [6]. Moreover, at the end of the 1970s, SAR focusing was one of the first applications for supercomputers. Wolf et al. present the assessment of implementing an SAR processor on a Cray-1 S Supercomputer [7]. Today, the implementation of quantum radars [8] and the use of quantum computers for further progress of SAR data processing and analysis is studied.

Other RS big data are generated from a multitude of sources, including ground and airborne sensors [e.g., unmanned aerial vehicles (UAVs)] [9], social media, machine-to-machine communications, and crowdsourcing. Meanwhile, planetary-scale applications in Earth science and environmental studies are further increasing the complexity of RS data. RS data can therefore be characterized by multisource, multiscale, high-dimensional, dynamic-state, and nonlinear characteristics [10]. Processing such large amounts of complex data necessitates rapid development in innovative computing technologies and creating novel tools for addressing data storage challenges and improving data processing workflows.

An increasing number of research groups have been working in the field of high-performance and cloud computing applied to RS, especially during the last few years [11], [12]. The GRSS is the right forum to foster bonds among these researchers and promote the use of these technologies by an ever-increasing

Digital Object Identifier 10.1109/MGRS.2022.3145478
Date of current version: 20 June 2022

community. The HDCRS WG was founded with these objectives. Through its dedicated website, HDCRS disseminates information, including activities organized by its members. IEEE Members can register as new members using the website.

The first activities of HDCRS were organized in 2021 and focus mainly on education and research promotion, with the goal of creating a community. The group encourages members to promote their related initiatives. In particular, HDCRS organized its first summer school at the University of Iceland from 31 May to 3 June 2021. The overall objective of the school was to give participants a comprehensive overview of current topics and methods in the field of HPC, machine learning (ML), and QC in RS. A second objective was to establish a venue for students and young professionals to network with senior researchers and professors who are world-renowned leaders in the field of RS, and work on the interdisciplinary research addressed by HDCRS.

THE FIRST ACTIVITIES OF HDCRS WERE ORGANIZED IN 2021 AND FOCUS MAINLY ON EDUCATION AND RESEARCH PROMOTION, WITH THE GOAL OF CREATING A COMMUNITY.

The first edition took place online due to the COVID-19 pandemic conditions. Prof. Jón Atli Benediktsson, rector of the University of Iceland, gave the opening remarks, summarizing the opportunities offered within the GRSS and its connection with the activities of the WG, which was presented by one of the chairs, Dr. Gabriele Cavallaro. The given lectures were organized into the following three thematic groups:

1) "From HPC to Quantum Paradigms in Earth Observation"
2) "Programming Graphics Processing Units and Accelerators with Directives"
3) "Scaling Machine Learning for RS Using Cloud Computing."

Out of 180 registrations from all over the world, the maximum number of attendees, 30, were admitted into the Zoom sessions and received access to computing resources. The rest attended via YouTube live streams of the Zoom sessions. Recordings of all the summer school lectures are available on the GRSS YouTube Channel.

HDCRS was happy to receive very favorable feedback for the summer school and is looking forward to organizing the second edition as a physical event at the University of Iceland, along with several social activities. Registrations will open on the HDCRS's website on 1 March 2022. It is envisioned that future editions of the summer school could be moved to other locations.

HDCRS has also organized two tutorials at the International Geoscience and Remote Sensing Symposium (IGARSS) conference. The first one, "Scalable ML With High Performance and Cloud Computing," provided a complete overview of supercomputing and cloud computing technologies for solving RS problems that require fast and highly scalable methods. The second tutorial, "From Big EO Data to Digital Twins: Hybrid Artificial Intelligence and Quantum-Based Paradigms," covered quantum information theory, quantum algorithms and computers, presented the first results, and analyzed main perspectives for Earth-observation (EO) applications.

A special session at the IGARSS 2021 conference was also organized by HDCRS. Papers in the most advanced areas exploiting new high-performance and distributed computing technologies and algorithms to expedite the processing and analysis of big RS data were collected. They included

- "Practice and Experience in Using Parallel and Scalable Machine Learning in Remote Sensing From HPC Over Cloud to Quantum Computing" [13]
- "Comparing Area-Based and Feature-Based Methods for Co-Registration of Multispectral Bands on GPU" [14]
- "An FPGA-Based Implementation of a Hyperspectral Anomaly Detection Algorithm for Real-Time Applications" [15]
- "Enhancing Large Batch Size Training of Deep Models for Remote Sensing Applications" [16]
- "Evolutionary Optimization of Neural Architectures in Remote Sensing Classification Problems." [17]

HDCRS will organize new special sessions on different topics in the future editions of IGARSS.

HDRCS RESEARCH TOPICS

There is an increasing number of applications that benefit from the amount of data acquired by the most affordable and widely available RS sensors. Some of them require processing in real time and most of them are complex, thus requiring high computational power. This requirement makes necessary the use of innovative computational approaches, from HPC platforms such as clusters, grids, or clouds, to accelerators such as GPUs or field-programmable gate arrays (FPGAs) or QC solutions, among others. The more adequate computing platform depends on the problem being solved as well as the environment where the problem needs to be solved. In some cases, for example, transferring data to supercomputers makes sense. In other cases, the problem is better solved in situ, using commodity hardware. In this section, a perspective of the potential and emerging challenges of applying HPC paradigms to RS problems is offered.

To solve a computational task, the first step is to split it into instructions that a processor can execute. The main objective is to process these instructions as fast as possible. This can be achieved in three different ways to make the processor 1) work harder (increase the raw power of the hardware, i.e., its clock speed on a single core, also referred to as *single-thread performance*), 2) work smarter (optimizing the task, use instruction-level parallelism and exploit caching, and so on), or 3) work in a team (more cores working in concert). Although the first two strategies formed the basis of the main computing trend in the first 50 years of hardware computing, the latter one is currently the main trend.

The semiconductor industry has been shrinking the technology to try to follow Moore's law [18]:

> … the number of transistors that can be inexpensively placed on integrated circuits is increasing exponentially, doubling approximately every two years … .
> —Gordon Moore, 1965

The result was that by doubling the density of semiconductor over integrated circuits, the single-thread performance constantly increased. This trend was also identified by Robert H. Dennard, who in 1974 predicted that the power density (i.e., power dissipated per unit area) of transistors would remain constant while their size would continue to decrease [19] (i.e., as the physical parameters of transistors reduce, they can be operated at lower voltage and thus at lower power). This meant that it was possible to constantly increase single-thread performance without raising power consumption.

Dennard scaling (also known as *MOSFET* scaling) started to reach its physical limits around 2004 to an extent that the voltage could not be scaled down as much as the gate's length of the transistor. This, along with a rise in leakage current, resulted in increased rather than constant power density (i.e., more heat generated, which has to be dissipated through cooling solutions, as an increase in temperature beyond a certain level results in unreliable functionality of the chip). As a consequence, beginning in early 2000, single-thread performance improvements started plateauing, as shown in Figure 1. This resulted in a unique situation in which Moore's law [18] was still holding, but the computing performance, in return, was no longer as substantial as before [20].

Novel hardware architectures, along with shifts in code paradigms, became the focus of the industry to continue the same trends. Expanding the number of logical cores in CPUs and shifting toward accelerators and coprocessors, which work on lower frequencies but have considerably higher amounts of cores than CPUs, proved to be the most significant move. The result was a mainstream shift of focus toward parallelization. Heterogeneous computing unifying different hardware architectures emerged as the most effective way to keep up with the need for ever-higher computing performance. In this context, the responsibility for reaching better computational performance is outsourced to software developers and programmers (i.e., algorithms need to be optimized to fully exploit new parallel computing environments) (see Figure 2).

FIGURE 1. The speakers at the 2021 HDCRS Summer School. Top row, from left: Gabriele Cavallaro, Mihai Datcu, Drew Bollinger, Jón Atli Benediktsson, and Manil Maskey. Bottom row, from left: Shubhankar Gahlot, Sergio Bernabé García, Muthukumaran Ramasubramanian, Iksha Gurung, and Carlos García Sánchez.

In general, the development of parallel and scalable codes for complex algorithms is complicated and error prone. It usually involves handling data slicing and distribution, task partition, message passing among distributed memory spaces and shared memory management for multicores, synchronization, and communication with low-level application programming interfaces [22]. Nevertheless, as was previously shown 10 years ago by Lee et al. [23], HPC and parallel programming are the only effective solutions that can address the computational challenges of data-intensive RS applications.

RESEARCH TOPICS

The essential concepts and principles, and the key techniques related to different computing technologies are elaborated on in the following sections. They describe how they RS applications are enhanced and provide future perspectives in the context of EO.

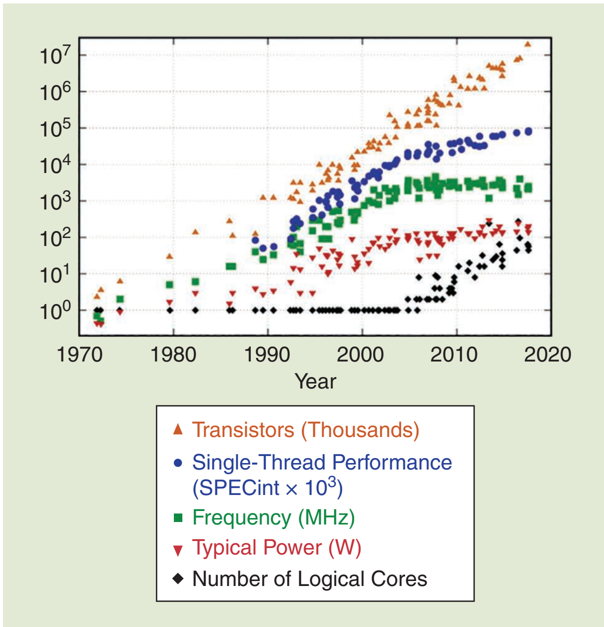

FIGURE 2. Forty-two years of microprocessor trend data [21]. The orange points: Moore's law trend. Circa 2003, the clock speed curve (blue points: single-thread performance) starts to flatten (i.e., the Dennard scaling breakdown). The green and red points: immediate consequences of the Dennard scaling breakdown; black points: from 2003, the era of parallelism begins (i.e., obtaining of processing speed up with many cores). SPECint: the integer performance testing component of the Standard Performance Evaluation Corporation test suit.

SUPERCOMPUTING

The action of solving processing tasks on a supercomputer is widely termed *supercomputing* and is synonymous with HPC. HPC is a multidisciplinary field of research that combines hardware technologies and architecture, operating systems, programming tools, software, and end-user problems and algorithms. It engages a class of electronic digital machines referred to as *supercomputers* to perform a wide array of computational problems or "applications" (alternatively, "workloads") as fast as is possible. A supercomputer is a mixture of shared- and distributed-memory systems. While in a shared-memory system (i.e., desktop computer, laptop), a number of CPU cores have access to a common, shared physical address space, in a distributed-memory system, each process is connected to exclusive local memory (i.e., no other process has direct access to it).

THIS REQUIREMENT MAKES NECESSARY THE USE OF INNOVATIVE COMPUTATIONAL APPROACHES, FROM HPC PLATFORMS SUCH AS CLUSTERS, GRIDS, OR CLOUDS, TO ACCELERATORS SUCH AS GPUS OR FPGAS OR QC SOLUTIONS, AMONG OTHERS.

Supercomputers have been used in various fields of research since the 1980s [24]. At that time, a vector architecture was the mainstream, and developers could improve the performance of programs by exploiting vector instructions. A vector instruction is *single-instruction multiple data*, which refers to the vector registers where multiple data reside. The first commercial supercomputer (i.e., the Cray-1 [25]) included eight registers, where each was a vector of 64 double-precision floating point numbers.

Single-thread exponential speed growth was the driving force of HPC in the first 25 years [21]. At first, each manufacturer of a distributed-memory system had its own library and set of functions that could do simple point-to-point communication as well as collective communication patterns like broadcasting. To simplify programming in network environments and realize component-based software architectures, many models and portable libraries have emerged as possible standards (i.e., a distributed component object model [26], parallel virtual machine [27], message passing interface (MPI) [28], and so forth).

MPI was released in 1994 and developed as a standard library of defined message passing. Since then, MPI has become extremely successful and been adopted by many different scientific applications for distributing their computations on distributed-memory clusters (e.g., hydrogeology, traffic simulation, weather forecasting, and so on [29]). MPI has become the de facto standard for parallel scientific computing, and they are the most mature method currently used in parallel programming.

Supercomputers have been widely used in RS applications to accelerate and scale the process of image mosaicking [30], [31], classification [32]–[37], object detection [38], [39], clustering [40]–[42], interband registration [43], superresolution [44], data fusion [16], compression [45], feature selection/extraction [46]–[48], spectral unmixing [49], data assimilation [50], and scalable-processing workflows [51]–[56]. In the context of HPC, there were also important efforts in academic journals and conferences, launching multiple special issues devoted to the processing and analysis of RS data [57]–[60].

The next generation of supercomputers (i.e., exascale supercomputers) will be used to model and simulate more complex and dynamic systems in higher resolution and with unprecedented fidelity (e.g., biological systems, molecular interactions of viruses, material design, and so forth). In the context of EO, exascale supercomputers will enable the development of a high-precision digital model of Earth (i.e., Destination Earth [61]). This will help analyze, with very high precision, the effects of climate change together with possible adaptation and mitigation strategies (e.g., to predict major environmental degradation and natural disasters with unprecedented fidelity and reliability).

CLOUD COMPUTING

Cloud computing is an overarching term that describes a category of on-demand computing services [62]. These services were initially offered by commercial companies such as Amazon, Microsoft, and Google. Now there are many new commercial and public cloud computing providers. The underlying principle behind cloud computing is the idea of providing access to storage, compute, and software "as a service," which may not be on premise. The common characteristics of cloud computing include

- *elasticity*: the ability to scale resources both up and down as needed
- *reliability*: implies that the service is available and works as intended
- *pay as you go*: only users pay for what they use
- *resource pooling*: allows a cloud provider to serve its users in a multitenant model
- *minimal management effort*: users can use and procure cloud services without much difficulty.

The concept of cloud computing is not new. Grid computing [63], which was introduced in the 1990s, included a type of parallel and distributed system that enabled the sharing of geographically distributed resources. The power of grid computing was enabled by the ability to dynamically scale up and down resources based on the user's need. The concept of grid computing evolved to solve large-scale processing workloads that required more than a single computer. Cloud computing automated some of the nuances of grid computing, specifically in the area of virtualization and on-demand scaling. Compared to the grid computing approach, which requires allocation of resources in advance, cloud computing is more attractive as real-time provisioning of resources is possible.

As cloud computing has advanced, the main services offered by many providers have evolved into three classes based on the abstraction level of the capability that they provide: 1) infrastructure as a service (IaaS), 2) platform as a service (PaaS), and 3) software as a service (SaaS) [62].

Figure 3 depicts the three layers, which shows the stacked organization from the infrastructure to the application layer. Each higher layer can utilize the services from the bottom layers.

IaaS uses virtualization technology to deliver computation, storage, and networking on demand. The cloud providers enable on-demand provisioning of servers, which can be used to develop applications. The users of IaaS will require system administration knowledge and usually have full control over the virtualized machine. Amazon Elastic Compute Cloud (http://aws.amazon.com/ec2/) is an example of IaaS.

PaaS is an environment where users can create customized solutions using tools and services that the platform provides. This layer is at a higher level of abstraction, which makes a cloud easily programmable. Often, a PaaS tool is a fully integrated development environment, that is, all the tools and services are a part of the PaaS service, which supports a complete lifecycle of building and deploying applications. Google App Engine is an example of PaaS.

SaaS is a complete cloud computing service model where the computing hardware, software, and a particular solution itself are provided by a vendor as a complete service offering. The services provided by this layer can be accessed by end users through browsers. For this reason, many users are increasingly shifting to online software services. The Aeronautical Reconnaissance Coverage Geographic Information System (ArcGIS) implementation on the cloud is an example of SaaS.

With the advances in sensor technology and highly competitive and vibrant space industry, the RS data are being collected at massive scale. Moreover, there are upcoming missions with higher spatial, spectral, and temporal resolution, which pose challenges for not only storing the data but also processing needs. To address these challenges, many agencies have already explored cloud computing as a viable solution. Cloud computing provides elasticity in storage and computing, which traditional data centers cannot support. Cloud computing also facilitates large-scale scientific processing enabled by the cloud-native services that are collocated with the data. During the last two decades, there has been an accelerated adoption of cloud computing within the RS community. This adoption trend can be observed in the number of periodicals by major RS research publishers that are related to cloud computing.

NASA has started migration of its Earth science data to the cloud computing environment (https://earthdata.nasa.gov/eosdis/cloud-evolution) to support the large data volume missions that will be launched in the near future. Toward that end, NASA has developed a generalized cloud-native ingest archive pipeline called *Cumulus* [64]. In the meantime, there are parallel efforts to train scientists to perform scientific analysis in cloud computing environments as it is more economical to perform analyses in the cloud than download a large amount of data from the cloud to on premise. Hence, cloud computing has also emerged as the analysis and processing platform for many applications [65]. In RS, there are many examples of data processing frameworks developed in cloud computing [66]–[70]. In fact, many new RS data products are being generated using cloud computing (https://earthdata.nasa.gov/learn/articles/hls-cloud-efforts). Cloud computing has also advanced the data storage and access techniques of RS data sets. Such advances have allowed dynamic data visualization and analysis, which are otherwise not possible [71], [72]. Finally, the development of end-to-end, RS-based situational awareness tools [73] are enabled by cloud-native services, which are capable of delivering reliable, on-demand needs.

IN THE CONTEXT OF EARTH OBSERVATION, EXASCALE SUPERCOMPUTERS WILL ENABLE THE DEVELOPMENT OF A HIGH-PRECISION DIGITAL MODEL OF EARTH.

With cloud computing, any researcher around the world is able to use a browser and open RS data to perform scientific research. RS has especially benefited from cloud computing, and many existing legacy applications have the potential to adapt to and take advantage of cloud capabilities. However, there are challenges in adopting the cloud. These challenges include security, evolving cloud-native services, multicloud portability, and the learning curve required to perform science experiments on the cloud.

SPECIALIZED HARDWARE COMPUTING

Numerous research efforts have been directed toward the incorporation of specialized hardware for accelerating RS-related applications during the last decade [74]–[76]. The emergence of specialized hardware devices such as FPGAs [77] or GPUs [78] have exhibited the potential to bridge the gap toward onboard and fast on-the-ground analysis of RS data. The small size and relatively low cost of these devices as compared to clusters or networks of computers makes them very appealing for parallel computing in general, and for RS

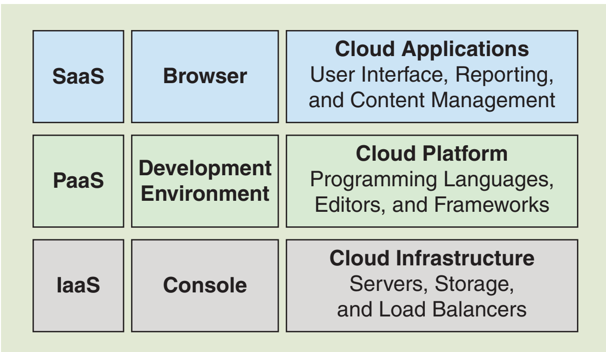

FIGURE 3. Cloud services—a layered view.

in particular. GPUs can also significantly increase the computational power of cluster-based systems and, today, they can be found in the most powerful nondistributed computer systems in the world (http://top500.org). In the case of FPGAs, their main advantage is configurability, although they are generally more expensive than GPUs (see Figure 4).

FPGAs have been consolidated as the standard choice for onboard RS image processing due to their programmable nature, dynamic reconfiguration capabilities, smaller size, weight, and power consumption as well as for the existence of radiation-hardened and radiation-tolerant FPGAs [79]–[82]. However, these devices are more expensive, physically larger, and often technology-generations behind in both performance and functionality than their commercial counterparts [79], [80]. For this reason, the current trend for small satellites is to use commercial off-the-shelf (COTS) onboard electronic devices. Moreover, commercial FPGAs based on static random-access memory are attracting attention because of their reconfiguration capabilities and low cost compared to application-specific integrated circuits [83]. Nonetheless, the use of COTS devices implies the necessity of applying mitigation techniques to increase the robustness of the application performance in environments exposed to radiation. In this sense, different radiation-hardened-by-design (RHBD) strategies have been developed over the years to protect FPGA-based designs against radiation [84]–[86], such as dual-modular-redundancy schemes for detecting errors and triple-modular-redundancy designs for error masking.

NASA HAS STARTED MIGRATION OF ITS EARTH SCIENCE DATA TO THE CLOUD COMPUTING ENVIRONMENT TO SUPPORT THE LARGE DATA VOLUME MISSIONS THAT WILL BE LAUNCHED IN THE NEAR FUTURE.

Although recent literature features plenty of works related to the utilization of FPGA devices for real-time onboard processing (including classification, detection, and spectral unmixing [88], [89] among many other processes such as hyperspectral image classification [90], [91]), the more significant advances have been achieved in the field of onboard compression. In fact, developing efficient compression solutions for space supposes a challenge: the employed algorithms must achieve the goal in terms of compression ratio while at the same time, they should have low complexity to be executed on the available hardware resources on board satellites and the required timing performance to meet mission requirements.

There is an immense quantity of contributions to the field of FPGA implementations for onboard data and image compression, both on COTS and RHBD devices. Of particular focus are those that follow the compression techniques proposed by the Consultative Committee for Space Data Systems (CCSDS), an international organization comprising the main space agencies in the world to define a common way for developing space data and information systems. Within these implementations, it is worth highlighting the works that implement the CCSDS 121.0-B-2 data compression standard [92], [93], which is based on Rice coding, onto space-qualified FPGAs as well as those that implement the CCSDS 123.0-B-1 lossless hyperspectral image-compression standard, both in COTS and RHBD FPGAs [82], [94]–[97].

Although GPUs had traditionally been limited to graphical operations, during the last decades they progressively evolved into highly parallel, multithreaded, many-core processors with tremendous computational speed and very high memory bandwidth [98]. In GPUs, more transistors are devoted to data processing than data caching and flow control. With the release of Nvidia's Compute Unified Device Architecture (CUDA) (http://developer.nvidia.com) in 2007 and OpenCL [99] in 2009, the programming model for GPUs was greatly simplified, introducing the possibility of including GPUs in many science and engineering applications. CUDA is an extension of the C programming language, offering the programming capabilities of GPUs for general-purpose computation. OpenCL was developed by a consortium and released in

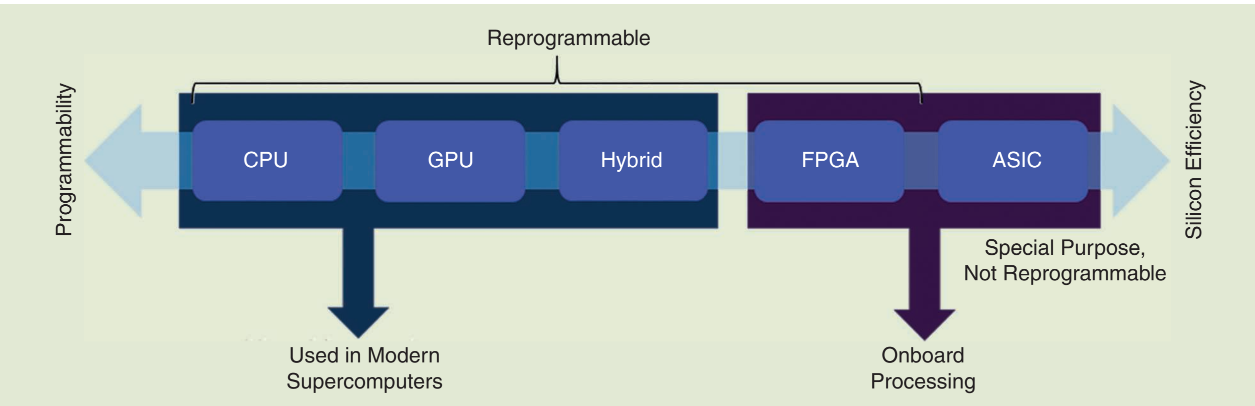

FIGURE 4. The programmability versus performance tradeoff of computing devices [87]. ASIC: application-specific integrated circuit.

2009. It aims at supporting more hardware and providing a standard for general-purpose parallel programming across CPUs, GPUs, and other processors [99]. Today, the combined features of general-purpose supercomputing, high parallelism, high memory bandwidth, and low cost makes a GPU-based computer an appealing alternative to a massively parallel system made up of only CPUs [75], [100].

The first developments in CUDA presented highly coupled and nonreusable GPU-parallel strategies. Many efforts were made for developing parallel programming templates [101] and libraries (https://docs.nvidia.com/cuda/) to simplify the programming task. The extraordinary evolution in this aspect during the last few years has motivated the extended use of GPUs for accelerating many different RS and, in particular, hyperspectral imaging-related tasks [74]–[76], [100], [102], [103]. These include registration [14], [104], segmentation [105], classification [76], or change detection [106], among others.

Based on the capability to execute thousands of threads in parallel, primitives such as the inner and outer products can perform better in the CUDA platform, so the ML and, in particular, deep learning algorithms formed by these primitives benefit from the computational capacity of CUDA [103]. For example, the convolutional neural network convolution, pooling, and activation calculation operations are readily portable to GPUs [107]. In this context, many tools have been developed to automatize the programming and execution of deep learning algorithms in GPU-based architectures, among which TensorFlow is the most popular option [108]. This has contributed to the extensive use of GPUs for deep learning applied to RS for many operations [109], [110] including, for example, object detection [111] or classification [112]–[116].

As explained previously, FPGAs and GPUs clearly help in processing RS data by accelerating computations and providing solutions for time-critical applications on board and on ground, which opens a wide variety of use cases related to Earth monitoring. Benefiting from them requires the careful selection of algorithms that better adapt to FPGA and GPU architectures. For the particular case of GPUs, many papers present algorithms and techniques adapted to them, as mentioned in the previous paragraphs, but GPUs are not being extensively exploited yet. More research is required for the development of new techniques, algorithms, and applications to exploit all the potentials for execution-time improvement that the wide variety of systems using GPUs offer.

EDGE COMPUTING

With the rapid advance in Internet of Things (IoT) technology, the number of network edge devices and amount of data generated by edge devices have shown explosive growth in recent years. Due to limited network communication capacity, the centralized processing mode in cloud computing may not be able to process massive amounts of data efficiently and quickly.

In 2013, the concept of edge computing was first mentioned by Ryan Lamothe of the Pacific Northwest National Laboratory. In 2016, Weisong Shi proposed that *edge computing* refers to the technologies computing at the edge of the network. This includes the processing of downstream and upstream data by cloud services and IoT services, respectively [117].

Generally, edge computing has two operation modes: 1) *binary offloading*, which refers to a deeply integrated or comparatively simple computing task that cannot be divided and has to run either directly on the edge device or offloaded to the cloud, and 2) *partial offloading*, which refers to a portion of the tasks originally located in the cloud data center that are allowed to be offloaded to the edge of the network. Through the two operation modes, edge computing can flexibly adjust the load of cloud and edge servers via offloading so as to realize the requirements of massive connection and low response delay of IoT devices. In certain cases, users can save more than 30% of the cost of computation, storage, and bandwidth. Mobile edge servers can also control the proximity between edge devices and terminal users so that they can track the real-time information of terminal users, such as action, location, and environment. In addition, mobile edge computing can protect privacy and enhance the security of mobile applications [118].

THERE IS AN IMMENSE QUANTITY OF CONTRIBUTIONS TO THE FIELD OF FPGA IMPLEMENTATIONS FOR ONBOARD DATA AND IMAGE COMPRESSION, BOTH ON COTS AND RHBD DEVICES.

Benefiting from the advantages of low latency, low power, and strong privacy, edge computing has attracted considerable attention from researchers, and it has been widely used in industrial fields such as autonomous driving environment monitoring, intelligent home virtual enhancement, medical and health industry production, and so on. For example, in the field of autonomous driving, a car does not need to send all the generated data to the cloud for processing. Most of the data are stored and calculated at the edge nodes (i.e., the car itself).

Although it is effective in reducing computing delay and power consumption, edge computing is also facing new challenges. First, limited by the computing capacity of edge devices, the accuracy of calculation results needs to be further improved. Second, most of the devices in edge computing are heterogeneous computing platforms, and the operating environment and data on each device are quite different. Therefore, it is challenging to deploy user applications in edge computing scenarios. In addition, as of yet, there are no comprehensive and uniform benchmarks for evaluating system performance.

Edge computing has been extensively used in various fields of RS. As the computing capacity of most edge devices is limited, the most common use of edge computing in RS applications is data preprocessing, which is able to mitigate transmission pressure and decrease computing cost in the cloud. In [119], a multiple Industrial IoT (IIoT) system architecture based on UAVs is proposed in which the RS images collected by sensors in the IIoT are directly transmitted to the UAVs for processing. Based on RS image analysis and the neural computation model, the authors in [120] built a forest ecotourism evaluation scheme and designed a cloud-based MEC model to construct efficient prediction scenarios [120]. In [121], the image recognition performance of a hierarchical discriminant analysis (HDA) algorithm was implemented by combining an edge computing environment with an HDA algorithm for early warnings of mountain fires.

With the increasing applications of edge computing in RS, there are many aspects that need to be further researched. First, the performance of edge equipment, the ability to collect RS information, and data processing need to be strengthened so as to promote the accuracy of the edge calculation result. Second, cloud-edge offloading strategies for RS need to be proposed to allocate computing resources more reasonably so as to reduce computing delay and power consumption in RS applications.

QC

At the beginning of the 1980s, Richard Feynman [122] observed that the numerical simulation of quantum mechanical systems required an exponentially growing—with the quantum mechanical system dimension—number of computational resources, such as CPU time and memory. This observation has led to the conclusion that, for the simulation of quantum mechanical systems, one should employ easily controllable quantum devices whose complexity can grow subexponentially with the growth of the quantum mechanical system dimension. Feynman named this easily controllable device a *quantum computer*. The first formal formulation of QC was proposed in 1985 by David Deutsch [123]. In 1992, Deutsch and Richard Jozsa proposed the first quantum algorithm that could outperform its classical counterpart [124]. In subsequent years, many other important quantum algorithms were proposed, such as Shor's algorithm for factoring integers [125], [126], Grover's search algorithm [127], and the Harrow–Hassidim–Lloyd algorithm for solving a linear system of equations [128].

Quantum computers can be understood as being analog and digital at the same time; analog because the state space of quantum devices during the computation process can be described by a set of continuous variables, and digital because the measurement outcome from a quantum computer can be expressed as a binary string. Quantum computers, as with most analog computers, are prone to errors. Due to its uncontrolled interaction with the environment, the state of a quantum computer can become distorted during the computation process. This phenomenon is called *quantum decoherence* [129]. Fortunately, the influence of decoherence can be reduced by the use of quantum-error-correcting codes. These codes employ multiple physical qubits to form a single logical qubit [130] and use the digital aspect of the quantum measurement to correct quantum errors.

Currently, quantum computers have reached noisy intermediate-scale quantum era [131]. This means that they consist of roughly 100 noisy qubits, and therefore, classical computers are unable to simulate them efficiently. Simultaneously, it is possible to perform only short quantum programs before the quantum state becomes so distorted that it is no longer useful. Hence, it is impossible to execute such algorithms as Shors' [125] and Grover's [127] using current quantum hardware.

Currently, two paradigms of QC are implemented in the hardware. The first one is universal gate-based QC, and the second is quantum annealing. Today, gate-based QC is mostly used to execute variational QC algorithms [132], a class of algorithms that uses a quantum computer as a coprocessor to execute computationally costly subroutines in which the value of a quantum observable for a particular state generated by a parametrized quantum circuit is estimated. In variational quantum algorithms, parameters of the quantum circuit are optimized in an iterative process using a classical optimization technique. Variational quantum algorithms have applications in combinatorial optimization problems, finding low-energy states of molecules, and in ML.

Quantum annealing [133] is a heuristic computation method that implements approximately the adiabatic QC model. This model enables finding good, approximate solutions to quadratic unconstrained binary optimization problems [134]. This is a class of computationally hard problems that find applications in logistics, scheduling, image processing, and ML, among others.

Even though the quantum advantage, that is, solving a particular computational task that is impossible to solve classically using a quantum computer, was claimed by Google [135] in 2019, current quantum computers have no practical applications as of yet. Fortunately, the field is progressing quickly, both in terms of algorithms and hardware development.

Quantum ML (*QML*) [136], [137] is a term that can encapsulate both the techniques of using quantum computers as ML subroutines during training and inference, or using quantum computers to help train classical classifiers. QML is currently a very active area of research that, hopefully, could enable building better models for a variety of ML tasks.

In the field of RS, there are particular applications of QC that have been developed recently. For example, in [138]–[141], QML algorithms such as support vector machines (SVMs) and neural networks are applied for classification of multispectral images. In [142], the authors use a quantum annealer to perform the following three tasks on hyperspectral data: classification using a variant of SVMs, band selection for classification, and boosting of classical classifiers. Outside of the applications to hyperspectral imaging, the authors of [143] proposed a classification method for SAR images using a hybrid quantum-classical neural network.

Today, the ESA considers QC and artificial intelligence taking center stage for the implementation of Digital Twin Earth (https://www.esa.int/Applications/Observing_the_Earth/Digital_Twin_Earth_quantum_computing_and_AI_take_centre_stage_at_ESA_s_Ph-week). Although QC technology concepts are broadening and growing in qubit capacities, their applications in RS and QML may have unexpected results. The analysis of data complexity and identification of optimal data embedding may open novel perspectives. For instance, signatures of satellite images could be encoded as quantum states and transformed using quantum kernels for classification. It might be feasible to encode a time-varying sequence of EO images on a quantum state and analyze it using a quantum computer to understand changes to the Earth's surface. But to achieve that, more efforts in both the theoretical development of quantum algorithms and quantum hardware design and production will have to be made to push the boundaries of what is possible to achieve with QC. An important aspect is the close collaboration with quantum computer developers and providing appropriate requirements [e.g., the European Quantum Industry Consortium (https://qt.eu/about-quantum-flagship/the-quantum-flagship-community/quic/)].

BLOCKCHAIN

Open data have become a significant vector in all of the services consumed today, as enormous quantities of data are quickly accessible. Most of the time, distributing and retrieving data are drained through mediators, which impose control and evaluation policies for reliability and integrity of the data. As connections between data owners and data consumers are generally maintained through a central authority for practical goals, thus limiting the actions of users, intermediary technologies are necessary to ensure trust among participants, data availability, data validity, and data integrity, all in a transparent way.

The advent of technological progress and evolution in open source and distributed ledger technologies (DLTs) has demonstrated that it is possible to develop systems that prioritize individual jurisdiction over centralized control. Distributed ledgers are collections of replicated, shared, and synchronized digital records that are stored across multiple geographically disseminated sites. A blockchain is an example of a DLT that is fundamentally an append-only, permanently verifiable data structure maintained by a set of nodes that do not fully trust each other. These nodes comply with a set of global states for an ordered collection of blocks, each containing multiple verification records (i.e., transactions). Each block is linked in a chain of blocks where the subsequent block, additionally, has a verification record of the previous block (i.e., a unique hash fingerprint). In this way, it is impossible to add new information to older blocks in the chain without changing subsequent blocks. Each node keeps replicas of the data and grants an execution order, thus producing an immutable log of ordered transactions within a distributed transaction management context.

Blockchains have manifested great promise in several fields like cryptocurrency (Bitcoin [144], Ethereum [145], and so forth), governance, land registration, justice, identity management, asset tracking, and the IoT, materializing in large-scale adoption as the result of solving limitations in previous systems. Blockchain technology has also started to evolve within the new space sector (i.e., Space 4.0) over a range of potential applications, from satellite communications to procurement. In a white paper, the ESA accentuated the relevance of assimilating blockchain into RS applications [146], supporting action automations through smart contracts and transfer of value without a pivotal authority. The data gathered via close-range sensors, e.g., IoT sensor networks or personal drones, can massively enrich EO applications in consistency and accuracy. The data owners can keep ownership, providing reliability through a blockchain solution.

Due to the verifiable and immutable nature of blockchain's technology, it can be used as a distributed database of digital fingerprints (e.g., mapping, cadastre, land registration [147], sharing continuously updated ML models [148], and so on.) As corruption can be a big challenge within administrative systems, the registration of land and real estate ownership using blockchain enhances transparency and accountability, bringing actors in control of their own data. The enormous repositories of data are transformed in intrinsically public open data by adopting blockchain and related technologies like the InterPlanetary File System [149], where no one controls data, anyone can access data, and anyone can audit the entire history of inputs. Novel blockchain protocols can also be used to precisely map physical world events in a temporal progression. For instance, cryptospatial coordinate (CSC) is an open and interoperable standard for location in Ethereum smart contracts. FOAM [150] is a CSC blockchain protocol that preserves geospatial data by validating the proof of location associated with the entry's specific time.

Blockchain technology brings important contributions in process management within complex systems, offering capabilities of managing massive patterns of transactions in any combination of two entities: human and device. SpaceChain builds an open source satellite network [151] in which satellites incorporate blockchain as an operating system and interface for decentralized applications to permit individuals to work on collaborative projects, with smart contracts on a space-based computing platform.

Blockchain can improve space communications and navigation, where the risk of transmission disruption can be eliminated by developing a decentralized, secure, and cognitive networking and computing infrastructure for deep space exploration [152]. A decentralized schema for verifying satellite locations in time through a type of proof-of-location protocol is proposed in [153]. The intent in using a permissioned blockchain is to facilitate scalability and trustless cooperation among satellite operators. The deployment and operation of small satellite constellations may encounter obstacles as satellite communications can be significantly delayed. In this case, occasionally, cryptographically secure, telemetry-based challenges are completed by satellites to verify the correctness of each other's position [154].

Blockchain solutions bring advantageous capabilities in data traceability and data reproducibility. A secure way of tracking down the changes made to the source data of the *Sentinel-2* satellite is considered in [155]. The authors proposed a system that captures each modification made to the original data set with the aim of being able to perform trace back and intermediate verification. In this design, data storage and data degradation problems still exist.

The synergy between blockchain and RS technologies is still fragile and sometimes divergent, but the dynamics of technological interaction sustains an evolving symbiosis and finds RS use cases in space asset tracking [156] and space communications as well as precision agriculture [157], among others. A blockchain-based, RS data-sharing model seems to be an applicable service that generates properties like immutability, decentralization, security, credibility, and collective maintenance, which are indispensable in communications among RS actors.

CONCLUSIONS

As the availability of sensors producing high amounts of RS data has increased, new applications of RS have emerged. The requirement of rapid and effective solutions for the processing of this massive data has led to the extended use of parallel execution. This article introduced the HDCRS, which is a WG of the GRSS, founded at the beginning of 2021, with the aim of promoting research, education, and job opportunities in the interdisciplinary field of RS and high-performance and disruptive computing. The key technologies involved in RS parallel computation—in particular, supercomputing, cloud computing, specialized hardware computing, QC, edge computing, and blockchain—were also presented. The most recent literature shows that new research is rapidly maturing at the intersection of the very different disciplines of RS and HPC.

AUTHOR INFORMATION

Gabriele Cavallaro (g.cavallaro@fz-juelich.de) received his B.Sc. and M.Sc. degrees in telecommunications engineering from the University of Trento, Italy, in 2011 and 2013, respectively, and his Ph.D. degree in electrical and computer engineering from the University of Iceland, Reykjavik, Iceland, in 2016. He is currently head of the AI and ML for Remote Sensing Simulation and Data Lab at the Jülich Supercomputing Centre, Wilhelm-Johnen-Strasse, 52428 Juelich, Germany. Since 2019, he has given lectures on scalable machine learning for remote sensing big data at the Institute of Geodesy and Geoinformation, the University of Bonn, Germany. He was the recipient of the IEEE GRSS Third Prize in the Student Paper Contest of the 2015 IEEE International Geoscience and Remote Sensing Symposium (Milan, Italy). He is chair of the High-Performance and Disruptive Computing in Remote Sensing Working Group of the IEEE Geoscience and Remote Sensing Society (GRSS) Earth Science Informatics Technical Committee. His research interests cover remote sensing data processing with parallel machine learning algorithms that scale on high-performance and distributed systems. He is a Member of IEEE.

Dora B. Heras (dora.blanco@usc.es) received her M.S. and Ph.D. degrees (cum laude) in physics from the University of Santiago de Compostela, Santiago, 15782, Spain, where she is an associate professor in the Department of Electronics and Computer Engineering and leads the remote sensing computing group. Since 2008, she has also been with Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Santiago de Compostela, 15782, Spain, and received the accreditation of full professor in 2020. Her publications have appeared in top-ranked journals and conferences. She is a reviewer for different *Journal Citation Reports* indexed journals in the areas of high performance computing and remote sensing. She also participated as a program committee member of different international conferences, in particular, the Euromicro International Conference on Parallel, Distributed and Network-Based Processing in 2021 and 2022. She is also chair of the High-Performance and Disruptive Computing in Remote Sensing Working Group of the IEEE Geoscience and Remote Sensing Society Earth Science Informatics Technical Committee. She has also been a member of the Euro-Par Conference Steering Committee since 2018 and has acted as cochair of the colocated workshops for the editions since 2017. Her research contributions cover a range of topics in the combined fields of image processing, remote sensing, machine learning, and high-performance computing. She is a Member of IEEE.

Zebin Wu (wuzb@njust.edu.cn) received his B.Sc. and Ph.D. degrees in 2003 and 2007, respectively, both in computer science and technology, from Nanjing University of Science and Technology. He is currently a professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, Jiangsu, 210094, China. He was a visiting scholar with the Grenoble Images Parole Signal Automatique-lab, Grenoble Institute of Technology, the University of Grenoble Alpes from August 2018 to September 2018; with the Department of Mathematics, the University of California, Los Angeles from August 2016 to September 2016 and July 2017 to August 2017; and with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, the University of Extremadura from June 2014 to June 2015. He is currently the principal investigator of three projects, funded by the National/Provincial Natural Science Foundation (NSFC) of China. He has also coordinated more than six scientific

research projects, supported by the Chinese Ministry of Science and Technology, the NSFC. He received the 2018 Young Teachers Award in Colleges and Universities of Henry Fok Education Foundation. He has authored 76 publications, including 30 *Journal Citation Reports* journal papers (22 in IEEE journals), and more than 30 peer-reviewed conference proceeding papers (25 in IEEE conferences). He served as vice chair of the IEEE Geoscience and Remote Sensing Society (GRSS) Nanjing Chapter during 2016–2020. He received the Best Reviewers Award of *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (*JSTARS*). He serves as associate editor of *JSTARS* and chair of the IEEE GRSS Nanjing Chapter. His research interests include hyperspectral image processing, parallel computing, and remotely sensed big data processing. He is a Senior Member of IEEE.

Manil Maskey (manil.maskey@nasa.gov) received his Ph.D. degree in computer science from the University of Alabama in Huntsville, Alabama. He is a senior research scientist with NASA, Marshall Space Flight Center, Huntsville, Alabama, 35808, USA. He also leads the Advanced Concepts team within the Inter Agency Implementation and Advanced Concepts. His career spans more than 20 years in academia, industry, and government. He chairs the IEEE Geoscience and Remote Sensing Society and Earth Science Informatics Technical Committee and leads machine learning activities for the NASA Earth Science Data Systems program. His research interests include computer vision, visualization, knowledge discovery, cloud computing, and data analytics. He is a Senior Member of IEEE.

Sebastián López (seblopez@iuma.ulpgc.es) received his M.S. degree from the University of La Laguna, Spain, and his Ph.D. degree from the University of Las Palmas de Gran Canaria, Spain, both in electronic engineering, in 2001 and 2006, respectively. He is an associate professor with the University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain, where he is currently involved in research activities with the Integrated Systems Design Division, Institute for Applied Microelectronics, Las Palmas de Gran Canaria, Las Palmas, 35001, Spain. He has coauthored more than 150 papers in international journals and conferences. He is also an active reviewer for various *Journal Citation Reports* journals and a program committee member of a variety of reputed international conferences. He has served as a program chair of the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing in 2014 and the SPIE Conference of High-Performance Computing in Remote Sensing from 2015 to 2018. He is currently an associate editor for *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *Remote Sensing*, and *Mathematical Problems in Engineering*. His research interests include real-time hyperspectral imaging, reconfigurable architectures, high-performance computing systems, and image and video processing. He is a Senior Member of IEEE.

Piotr Gawron (gawron@camk.edu.pl) received his magister title in computer science from the Silesian University of Technology, Gliwice, Poland, his doctoral degree in technical sciences from the Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Warsaw, Poland, and his habilitation doctoral degree from the faculty of Automatic Control, Electronics and Computer Science from the Silesian University of Technology in 2003, 2008, and 2014, respectively. He is leader of the Scientific Computing and Information Technology Group and an institute professor at The Particle Astrophysics Science and Technology Centre (AstroCeNT) International Research Agenda, Nicolaus Copernicus Astronomical Center, Polish Academy of Sciences, Warszawa, 00-614, Poland. For 18 years, he was a member of the Quantum Systems of Informatics Group at the Institute of Theoretical and Applied Informatics of the Polish Academy Sciences in Gliwice, Poland. He has been involved in quantum computer science research since the fourth year of his university studies. Previously, he was engaged in research on quantum games, quantum walks, simulation of noisy quantum computers, quantum programming languages, quantum control, numerical shadows, and tensor networks. Currently, he is studying applicability of quantum machine learning for Earth-observation imagery data processing, applications of quantum and classical machine learning techniques for gravitational wave, and dark matter detection.

Mihai Coca (mihai.coca@mta.ro) received his B.S. and M.S. degrees in computer science and information security from Military Technical Academy “Ferdinand I,” in 2015 and 2017, respectively. He is currently working toward his Ph.D. degree in electronics, telecommunications, and information technology at University Politechnica of Bucharest, Bucharest, Romania, where he leads the Computer Science and Cyber Security Laboratory at Military Technical Academy “Ferdinand I,” Bucureşti, 050141, Romania. He joined CEO-SpaceTech as a research assistant in 2017. Since 2015, he has been an associate professor with Military Technical Academy “Ferdinand I,” covering subjects such as the architecture of computer systems, and microprocessor systems. His current research interests include Earth-observation multispectral and synthetic aperture radar processing based on temporal analysis in image time series, change detection, and anomaly detection. His supplementary research interests include decentralized and collaborative artificial intelligence based on blockchain technologies and machine learning algorithms for embedded systems.

Mihai Datcu (mihai.datcu@dlr.de) is a senior scientist and data intelligence and knowledge discovery research group leader with the Remote Sensing Technology Institute of German Aerospace Center (DLR), Wessling, 82234, Germany, and a professor with the Department of Applied Electronics and Information Engineering, Faculty of Electronics, Telecommunications and Information Technology, University Politechnica Bucharest, Bucureşti, 061071, Romania. From 1992 to 2002, he had an Invited Professor Assignment with the Swiss Federal Institute of Technology, ETH Zurich, Switzerland. From 2005 to 2013, he was a professor holder of the DLR-CNES Chair at ParisTech, Paris Institute of Technology, Telecom Paris. He is involved in artificial intelligence (AI) and big data from space; with European, the European Space Agency (ESA), NASA, and national research programs and projects. He is a member of the ESA's Big Data From Space Working Group and a visiting professor with the ESA's ϕ-Lab. He is involved in quantum machine learning research and quantum radar signal processing. He has been a co-organizer of a series of summer schools at UPB on quantum theory and methods and is a member of the European Quantum Industry Consortium. He is involved in collaboration with NASA on applications of quantum annealing. In 2006, he received the Best Paper Award and the IEEE Geoscience and Remote Sensing Society Prize. In 2008, he was awarded the National Order of Merit with the rank of Knight "for outstanding international research results" by the president of Romania. In 2017, he was presented the Blaise Pascal Chairs "for international recognition in the field of data science in Earth observation," and in 2018, the Ad Astra Award for Excellence in Science. His research interests include explainable and physics-aware AI, smart sensors design, and quantum machine learning with applications in Earth observation. He is a Fellow of IEEE.

REFERENCES

[1] D. Manil, "Aerial photography," in *The Focal Encyclopedia of Photography*, Boston, MA, USA: Focal Press, 2007. [Online]. Available: https://doi.org/10.1016/b978-0-240-80740-9.50087-8

[2] N. M. Short, "The Landsat tutorial workbook: Basics of satellite remote sensing," Washington, D.C: National Aeronautics and Space Administration, Scientific and Technical Information Branch, Washington, D.C: National Aeronautics and Space Administration, Scientific and Technical Information Branch, 1982. [Online]. Available: https://www.worldcat.org/title/landsat-tutorial-workbook-basics-of-satellite-remote-sensing/oclc/8711568

[3] C. E. Woodcock et al., "Free access to Landsat imagery," *Science*, vol. 320, no. 5879, p. 1011, 2008, doi: 10.1126/science.320.5879.1011a.

[4] M. A. Wulder, J. G. Masek, W. B. Cohen, T. R. Loveland, and C. E. Woodcock, "Opening the archive: How free data has enabled the science and monitoring promise of Landsat," *Remote Sens. Environment*, vol. 122, pp. 2–10, Jul. 2012, doi: 10.1016/j.rse.2012.01.010.

[5] J. Aschbacher, "ESA's earth observation strategy and Copernicus," in *Satellite Earth Observations and Their Impact on Society and Policy*, M. Onoda and O. Young, Eds. Singapore: Springer-Verlag, 2017, pp. 81–86.

[6] L. J. Cutrona, E. N. Leith, L. J. Porcello, and W. E. Vivian, "On the application of coherent optical processing techniques to synthetic-aperture radar," *Proc. IEEE*, vol. 54, no. 8, pp. 1026–1032, 1966, doi: 10.1109/PROC.1966.4987.

[7] M. L. Wolf, D. J. Lewis, and D. G. Corr, "Synthetic aperture radar processing on a CRAY-1 S supercomputer," *Telematics Inform.*, vol. 2, no. 4, pp. 321–330, 1985, doi: 10.1016/0736-5853(85)90041-3.

[8] S. Barzanjeh, S. Pirandola, D. Vitali, and J. M. Fink, "Microwave quantum illumination using a digital receiver," *Sci. Adv.*, vol. 6, no. 19, p. eabb0451, 2020, doi: 10.1126/sciadv.abb0451.

[9] X. Liao, Y. Zhang, F. Su, H. Yue, Z. Ding, and J. Liu, "UAVs surpassing satellites and aircraft in remote sensing over China," *Int. J. Remote Sens.*, vol. 39, no. 21, pp. 7138–7153, 2018, doi: 10.1080/01431161.2018.1515511.

[10] M. Reichstein, *et al.*, "Deep learning and process understanding for data-driven earth system science," *Nature*, vol. 566, pp. 195–204, Feb. 2019, doi: 10.1038/s41586-019-0912-1.

[11] A. J. Plaza and C.-I. Chang, *High Performance Computing in Remote Sensing*. London, U.K.: Chapman & Hall, 2007.

[12] L. Wang, Y. Jining, and Y. Ma, *Cloud Computing in Remote Sensing*. London, U.K.: Chapman & Hall, 2019.

[13] M. Riedel, G. Cavallaro, and J. A. Benediktsson, "Practice and experience in using parallel and scalable machine learning in remote sensing from HPC over cloud to quantum computing," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 1571–1574, doi: 10.1109/IGARSS47720.2021.9554656.

[14] A. Ordóñez, D. B. Heras, and F. Argüello, "Comparing area-based and feature-based methods for co-registration of multispectral bands on GPU," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 1575–1578, doi: 10.1109/IGARSS47720.2021.9554152.

[15] M. Diaz, R. Guerra, S. Lopez, J. Caba, and J. Barba, "An FPGA-based implementation of a hyperspectral anomaly detection algorithm for real-time applications," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 1579–1582, doi: 10.1109/IGARSS47720.2021.9554801.

[16] R. Sedona, C. Paris, G. Cavallaro, L. Bruzzone, and M. Riedel, "A high-performance multispectral adaptation GAN for harmonizing dense time series of landsat-8 and sentinel-2 images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10,134–10,146, Sep. 2021, doi: 10.1109/JSTARS.2021.3115604.

[17] D. Coquelin, R. Sedona, M. Riedel, and M. Götz, "Evolutionary optimization of neural architectures in remote sensing classification problems," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 1587–1590, doi: 10.1109/IGARSS47720.2021.9554309.

[18] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics*, vol. 38, pp. 114–117, Apr. 19, 1965.

[19] R. H. Dennard, F. H. Gaensslen, H. N. Yu, V. L. Rideout, E. Bassous, and A. R. Leblanc, "Design of ion-implanted MOSFET's with very small physical dimensions," *IEEE J. Solid-State Circuits*, vol. 9, no. 5, pp. 256–268, 1974, doi: 10.1109/JSSC.1974.1050511.

[20] M. Bohr, "A 30 year retrospective on Dennard's MOSFET scaling paper," *IEEE Solid-State Circuits Soc. Newslett.*, vol. 12, no. 1, pp. 11–13, 2007, doi: 10.1109/N-SSC.2007.4785534.

[21] K. Rupp, "42 years of microprocessor trend data," 2018. karlrupp.net. https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

[22] Y. Ma, L. Wang, D. Liu, P. Liu, J. Wang, and J. Tao, "Generic parallel programming for massive remote sensing data processing," in *Proc. 2012 IEEE Int. Conf. Cluster Comput.*, pp. 420–428, doi: 10.1109/CLUSTER.2012.51.

[23] C. A. Lee, S. D. Gasster, A. Plaza, C.-I. Chang, and B. Huang, "Recent developments in high performance computing for remote sensing: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 508–527, 2011, doi: 10.1109/JSTARS.2011.2162643.

[24] M. Geshi, *The Art of High Performance Computing for Computational Science*, vol. 1. Singapore: Springer-Verlag, 2019.

[25] R. M. Russell, "The CRAY-1 computer system," *Commun. ACM*, vol. 21, no. 1, pp. 63–72, 1978, doi: 10.1145/359327.359336.

[26] N. Brown and C. Kindel, "Distributed component object model protocol -- DCOM/1.0," Internet Engineering Task Force, Fremont, CA, USA, Nov. 1998. [Online]. Available: https://datatracker.ietf.org/doc/draft-brown-dcom-v1-spec/

[27] V. S. Sunderam, "PVM: A framework for parallel distributed computing," *Concurrency, Pract. Experience*, vol. 2, no. 4, pp. 315–339, 1990, doi: 10.1002/cpe.4330020404.

[28] "MPI: A message-passing interface standard," MPI Forum, USA, Tech. Rep., 1994. [Online]. Available: http://www.netlib.org/utk/people/JackDongarra/PAPERS/059_1994_mpi-a-message-passing-interface-standard.pdf

[29] A. Skjellum, E. Lusk, and W. Gropp, "Early applications in the message-passing interface (MPI)," *Int. J. Supercomput. Appl. High Perform. Comput.*, vol. 9, no. 2, pp. 79–94, 1995, doi: 10.1177/109434209500900202.

[30] D. S. Katz et al., "Astronomical image mosaicking on a grid: Initial experiences," Jan. 2006, pp. 69–88. [Online]. Available: https://www.isi.edu/people/carl/publications/astronomical_image_mosaicking_grid_initial_experiences

[31] Y. Wang, Y. Ma, P. Liu, D. Liu, and J. Xie, "An optimized image mosaic algorithm with parallel IO and dynamic grouped parallel strategy based on minimal spanning tree," in *Proc. 2010 9th Int. Conf. Grid Cloud Comput.*, pp. 501–506, doi: 10.1109/GCC.2010.103.

[32] D. Valencia, A. Plaza, P. Cobo, and J. Plaza, "Parallel processing of high-dimensional remote sensing images using cluster computer architectures," *Int. J. Comput. Appl.*, vol. 14, no. 1, pp. 23–34, Jan. 2007.

[33] Y.-L. Chang, Z.-M. Chen, J.-N. Liu, L. Chang, and J. P. Fang, "Parallel K-dimensional tree classification based on semi-matroid structure for remote sensing applications," in *Proc. SPIE Satellite Data Compression, Commun., Process. VI*, vol. 7810, B. Huang, A. J. Plaza, J. Serra-Sagristà, C. Lee, Y. Li, and S.-E. Qian, Eds. Bellingham, WA, USA: SPIE, 2010, pp. 192–199, doi: 10.1117/12.862715.

[34] G. Cavallaro, M. Riedel, M. Richerzhagen, J. A. Benediktsson, and A. Plaza, "On understanding big data impacts in remotely sensed image classification using support vector machine methods," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 10, pp. 4634–4646, 2015, doi: 10.1109/JSTARS.2015.2458855.

[35] R. Latifovic, D. Pouliot, and I. Olthof, "Circa 2010 land cover of Canada: Local optimization methodology and product development," *Remote Sens.*, vol. 9, no. 11, p. 1098, 2017, doi: 10.3390/rs9111098.

[36] R. Sedona, G. Cavallaro, J. Jitsev, A. Strube, M. Riedel, and J. A. Benediktsson, "Remote sensing big data classification with high performance distributed deep learning," *Remote Sens.*, vol. 11, no. 24, p. 3056, 2019, doi: 10.3390/rs11243056.

[37] R. Sedona, G. Cavallaro, J. Jitsev, A. Strube, M. Riedel, and M. Book, "Scaling up a multispectral resnet-50 to 128 GPUs," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2020, pp. 1058–1061, doi: 10.1109/IGARSS39084.2020.9324237.

[38] B. Gokaraju, S. Bhushan, V. Anantharaj, A. C. Turlapaty, and D. A. Doss, "Comprehensive review of evolution of satellite sensor specifications against speedup performance of pattern recognition algorithms in remote sensing," in *Proc. 2015 IEEE Appl. Imagery Pattern Recognit. Workshop (AIPR)*, pp. 1–8, doi: 10.1109/AIPR.2015.7444540.

[39] U. Bhangale, S. S. Durbha, R. L. King, N. H. Younan, and R. Vatsavai, "High performance GPU computing based approaches for oil spill detection from multi-temporal remote sensing data," *Remote Sens. Environment*, vol. 202, pp. 28–44, Dec. 2017, doi: 10.1016/j.rse.2017.03.024.

[40] D. Petcu, D. Zaharie, S. Panica, A. S. Hussein, A. Sayed, and H. El-Shishiny, "Fuzzy clustering of large satellite images using high performance computing," in *Proc. SPIE High-Perform. Comput. Remote Sens.*, vol. 8183, B. Huang and A. J. Plaza, Eds. Bellingham, WA, USA: SPIE, 2011, pp. 11–28, doi: 10.1117/12.898281.

[41] M. Götz, C. Bodenstein, and M. Riedel, "HPDBSCAN: Highly parallel DBSCAN," in *Proc. 2015 MLHPC, Mach. Learn. High-Perform. Comput. Environments - Held Conjunction SC Int. Conf. High Perform. Comput., Netw., Storage Analysis*, pp. 1–10, doi: 10.1145/2834892.2834894.

[42] S. Sreepathi, J. Kumar, R. T. Mills, F. M. Hoffman, V. Sripathi, and W. W. Hargrove, "Parallel multivariate Spatio-temporal clustering of large ecological datasets on hybrid supercomputers," in *Proc. 2017 IEEE Int. Conf. Cluster Comput. (CLUSTER)*, pp. 267–277, doi: 10.1109/CLUSTER.2017.88.

[43] T. Kim, M. Choi, and T. Chae, "Parallel processing with MPI for inter-band registration in remote sensing," in *Proc. 2011 IEEE 17th Int. Conf. Parallel Distrib. Syst.*, pp. 1021–1025, doi: 10.1109/ICPADS.2011.95.

[44] R. Zhang, G. Cavallaro, and J. Jitsev, "Super-resolution of large volumes of sentinel-2 images with high performance distributed deep learning," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2020, pp. 617–620, doi: 10.1109/IGARSS39084.2020.9323734.

[45] L. Pesquer, A. Cortés, I. Serral, and X. Pons, "Geostatistical analysis of Landsat-TM lossy compression images in a high-performance computing environment," in *Proc. High-Perform. Comput. Remote Sens.*, 2011, p. 818,307, doi: 10.1117/12.896418.

[46] Y.-L. Chang, K.-S. Chen, B. Huang, W.-Y. Chang, J. A. Benediktsson, and L. Chang, "A parallel simulated annealing approach to band selection for high-dimensional remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 579–590, 2011, doi: 10.1109/JSTARS.2011.2160048.

[47] M. Gotz, G. Cavallaro, T. Geraud, M. Book, and M. Riedel, "Parallel computation of component trees on distributed memory machines," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 11, pp. 2582–2598, 2018, doi: 10.1109/TPDS.2018.2829724.

[48] S. Gazagnes and M. H. F. Wilkinson, "Distributed connected component filtering and analysis in 2D and 3D tera-scale data sets," *IEEE Trans. Image Process.*, vol. 30, pp. 3664–3675, Mar. 2021, doi: 10.1109/TIP.2021.3064223.

[49] A. C. Toma, S. Panica, D. Zaharie, and D. Petcu, "Computational challenges in processing large hyperspectral images," in *Proc. 2012 5th Romania Tier 2 Federation Grid, Cloud High Perform. Comput. Sci. (RQLCG)*, pp. 111–114.

[50] Y. Wang, Y. Jung, T. A. Supinie, and M. Xue, "A hybrid MPI–OPENMP parallel algorithm and performance analysis for an ensemble square root filter designed for multiscale observations," *J. Atmospheric Ocean. Technol.*, vol. 30, no. 7, pp. 1382–1397, Jul. 1, 2013, doi: 10.1175/JTECH-D-12-00165.1.

[51] L. Wang, Y. Ma, A. Y. Zomaya, R. Ranjan, and D. Chen, "A parallel file system with application-aware data layout policies for massive remote sensing image processing in digital earth," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 6, pp. 1497–1508, 2015, doi: 10.1109/TPDS.2014.2322362.

[52] K. R. Kurte, U. M. Bhangale, S. S. Durbha, R. L. King, and N. H. Younan, "Accelerating big data processing chain in image information mining using a hybrid HPC approach," in *Proc. 2016 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 7597–7600, doi: 10.1109/IGARSS.2016.7730981.

[53] U. M. Bhangale, K. R. Kurte, S. S. Durbha, R. L. King, and N. H. Younan, "Big data processing using HPC for remote sensing disaster data," in *Proc. 2016 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 5894–5897, doi: 10.1109/IGARSS.2016.7730540.

[54] F. Huang, J. Zhou, J. Tao, X. Tan, S. Liang, and J. Cheng, "PMODTRAN: A parallel implementation based on MODTRAN for massive remote sensing data processing," *Int. J. Digit. Earth*, vol. 9, no. 9, pp. 819–834, 2016, doi: 10.1080/17538947.2016.1144800.

[55] R. Cresson and G. Hautreux, "A generic framework for the development of geospatial processing pipelines on clusters," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 11, pp. 1706–1710, 2016, doi: 10.1109/LGRS.2016.2605138.

[56] O. Melet, A. Masse, Y. Ott, and P. Lassalle, "A new architecture paradigm for image processing pipeline applied to massive remote sensing data production," in *Proc. SPIE Image Signal Process. Remote Sens. XXIV*, vol. 10,789, L. Bruzzone and F. Bovolo, Eds. Bellingham, WA, USA: SPIE, 2018, pp. 136–142, doi: 10.1117/12.2325700.

[57] A. Plaza, Q. Du, Y. L. Chang, and R. L. King, "Foreword to the special issue on high performance computing in earth observation and remote sensing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 503–507, Sep. 2011, doi: 10.1109/JSTARS.2011.2163551.

[58] G. Lemoine and M. Giovalli, "Geo-correction of high-resolution imagery using fast template matching on a GPU in emergency mapping contexts," *Remote Sens.*, vol. 5, no. 9, pp. 4488–4502, 2013, doi: 10.3390/rs5094488.

[59] R. Zhang, G. Zhou, G. Zhang, X. Zhou, and J. Huang, "RPC-based orthorectification for satellite images using FPGA," *Sensors*, vol. 18, no. 8, p. 2511, 2018, doi: 10.3390/s18082511.

[60] "Front matter," in *Proc. SPIE High-Perform. Comput. Geosci. Remote Sens. VIII*, vol. 10792, B. Huang, S. López, and Z. Wu, Eds. Bellingham, WA, USA: SPIE, 2018, pp. 123–130, doi: 10.1117/12.2519955.

[61] N. Stefano and C. Max, "Destination earth: Ecosystem architecture description," European Commission, Brussels, Belgium, JRC Tech Rep. KJ-NA-30646-EN-N, 2021.

[62] P. Mell and T. Grance, "The NIST definition of cloud computing," NIST, Gaithersburg, MD, USA, Ver. 15, 2009. [Online]. Available: http://csrc.nist.gov/groups/SNS/cloud-computing/

[63] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Proc. 2008 Grid Comput. Environments Workshop*, pp. 1–10, doi: 10.1109/GCE.2008.4738445.

[64] R. Ramachandran, K. Baynes, K. Murphy, A. Jazayeri, I. Schuler, and D. Pilone, "CUMULUS: Nasa's cloud based distributed active archive center prototype," in *Proc. 2017 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 369–372, doi: 10.1109/IGARSS.2017.8126972.

[65] C. Dobre and F. Xhafa, "Parallel programming paradigms and frameworks in big data era," *Int. J. Parallel Program.*, vol. 42, no. 5, pp. 710–738, 2013, doi: 10.1007/s10766-013-0272-7.

[66] D. Mandl, "Matsu: An elastic cloud connected to a sensorweb for disaster response," 2011. [Online]. Available: https://www.semanticscholar.org/paper/Matsu%3A-An-Elastic-Cloud-Connected-to-a-SensorWeb-Mandl/b0531846e21874d4a33ff6c4f8f33a8b444c0ff7

[67] Z. Wu, Y. Li, A. Plaza, J. Li, F. Xiao, and Z. Wei, "Parallel and distributed dimensionality reduction of hyperspectral data on cloud computing architectures," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 6, pp. 2270–2278, 2016, doi: 10.1109/JSTARS.2016.2542193.

[68] J. Sun et al., "An efficient and scalable framework for processing remotely sensed big data in cloud computing environments," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4294–4308, 2019, doi: 10.1109/TGRS.2018.2890513.

[69] V. A. A. Quirita et al., "A new cloud computing architecture for the classification of remote sensing data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 2, pp. 409–416, 2017, doi: 10.1109/JSTARS.2016.2603120.

[70] K. Bugbee et al., "Advancing open science through innovative data system solutions: The joint ESA-NASA multi-mission algorithm and analysis platform (MAAP)'s data ecosystem," in *Proc. 2020 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 3097–3100, doi: 10.1109/IGARSS39084.2020.9323731.

[71] G. T. Stano, Y. Wu, N. R. Selvaraj, M. Maskey, and A. Kulkarni, "The field campaign explorer," in *Proc. 2021 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 3257–3260, doi: 10.1109/IGARSS47720.2021.9553677.

[72] M. Maskey et al., "Visualizing, exploring, and communicating environmental effects of COVID-19 using earth observation dashboard," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 1370–1373, doi: 10.1109/IGARSS47720.2021.9553461.

[73] M. Maskey et al., "Deepti: Deep-learning-based tropical cyclone intensity estimation system," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4271–4281, Jul. 2020, doi: 10.1109/JSTARS.2020.3011907.

[74] E. Christophe, J. Michel, and J. Inglada, "Remote sensing processing: From multicore to GPU," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 643–652, 2011, doi: 10.1109/JSTARS.2010.2102340.

[75] C. Gonzalez, S. Sánchez, A. Paz, J. Resano, D. Mozos, and A. Plaza, "Use of FPGA or GPU-based architectures for remotely sensed hyperspectral image processing," *Integration*, vol. 46, no. 2, pp. 89–103, 2013, doi: 10.1016/j.vlsi.2012.04.002.

[76] A. Yusuf and S. Alawneh, "A survey of GPU implementations for hyperspectral image classification in remote sensing," *Can. J. Remote Sens.*, vol. 44, no. 5, pp. 532–550, 2018, doi: 10.1080/07038992.2018.1559725.

[77] S. Hauck, "The roles of FPGAs in reprogrammable systems," *Proc. IEEE*, vol. 86, no. 4, pp. 615–638, 1998, doi: 10.1109/5.663540.

[78] E. Lindholm, J. Nickolls, S. Oberman, and J. Montrym, "NVIDIA Tesla: A unified graphics and computing architecture," *IEEE Micro*, vol. 28, no. 2, pp. 39–55, 2008, doi: 10.1109/MM.2008.31.

[79] S. Lopez, T. Vladimirova, C. Gonzalez, J. Resano, D. Mozos, and A. Plaza, "The promise of reconfigurable computing for hyperspectral imaging onboard systems: A review and trends," *Proc. IEEE*, vol. 101, no. 3, pp. 698–722, 2013, doi: 10.1109/JPROC.2012.2231391.

[80] A. D. George and C. M. Wilson, "Onboard processing with hybrid and reconfigurable computing on small satellites," *Proc. IEEE*, vol. 106, no. 3, pp. 458–470, 2018, doi: 10.1109/JPROC.2018.2802438.

[81] Q. Du and R. Nekovei, "Fast real-time onboard processing of hyperspectral imagery for detection and classification," *J. Real-Time Image Process.*, vol. 4, no. 3, pp. 273–286, 2009, doi: 10.1007/s11554-008-0106-9.

[82] M. Orlandić, J. Fjeldtvedt, and T. A. Johansen, "A parallel FPGA implementation of the CCSDS-123 compression algorithm," *Remote Sens.*, vol. 11, no. 6, p. 673, 2019, doi: 10.3390/rs11060673.

[83] L. A. Aranda, A. Sánchez, F. Garcia-Herrero, Y. Barrios, R. Sarmiento, and J. A. Maestro, "Reliability analysis of the SHyLoC CCSDS123 IP core for lossless hyperspectral image compression using COTS FPGAS," *Electronics*, vol. 9, no. 10, p. 1681, 2020, doi: 10.3390/electronics9101681.

[84] L. A. Aranda, P. Reviriego, and J. A. Maestro, "Toward a fault-tolerant star tracker for small satellite applications," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 5, pp. 3421–3431, 2020, doi: 10.1109/TAES.2020.2971289.

[85] J. A. Hogan, R. J. Weber, and B. J. LaMeres, "Reliability analysis of field-programmable gate-array-based space computer architectures," *J. Aerosp. Inform. Syst.*, vol. 14, no. 4, pp. 247–258, 2017, doi: 10.2514/1.I010481.

[86] S. Banteywalu, B. Khan, V. De Smedt, and P. Leroux, "A novel modular radiation hardening approach applied to a synchronous buck converter," *Electronics*, vol. 8, no. 5, p. 513, 2019, doi: 10.3390/electronics8050513.

[87] J. de Fine Licht, M. Besta, S. Meierhans, and T. Hoefler, "Transformations of high-level synthesis codes for high-performance computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 5, pp. 1014–1029, 2021, doi: 10.1109/TPDS.2020.3039409.

[88] T. G. Cervero et al., "A scalable and dynamically reconfigurable FPGA-based embedded system for real-time hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2894–2911, 2015, doi: 10.1109/JSTARS.2014.2347075.

[89] C. Gonzalez, J. Resano, A. Plaza, and D. Mozos, "FPGA implementation of abundance estimation for spectral unmixing of hyperspectral data using the image space reconstruction algorithm," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 1, pp. 248–261, 2012, doi: 10.1109/JSTARS.2011.2171673.

[90] Z. Zheng, Y. Zhong, A. Ma, and L. Zhang, "FPGA: Fast patch-free global learning framework for fully end-to-end hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 8, pp. 5612–5626, 2020, doi: 10.1109/TGRS.2020.2967821.

[91] S. Liu, R. S. W. Chu, X. Wang, and W. Luk, "Optimizing cnn-based hyperspectral image classification on FPGAs," in *Proc. 15th Int. Symp., ARC 2019*, Darmstadt, Germany, Apr. 9–11, 2019, pp. 17–31. [Online]. Available: https://www.springerprofessional.de/en/applied-reconfigurable-computing/16601678

[92] Y. Barrios, A. J. Sánchez, L. Santos, and R. Sarmiento, "SHyLoC 2.0: A versatile hardware solution for on-board data and hyperspectral image compression on future space missions," *IEEE Access*, vol. 8, pp. 54,269–54,287, Mar. 2020, doi: 10.1109/ACCESS.2020.2980767.

[93] N. Kranitis, I. Sideris, A. Tsigkanos, G. Theodorou, A. Paschalis, and R. Vitulli, "Efficient field-programmable gate array implementation of CCSDS 121.0-b-2 lossless data compression algorithm for image compression," *J. Appl. Remote Sens.*, vol. 9, no. 1, p. 097499, 2015, doi: 10.1117/1.JRS.9.097499.

[94] A. Tsigkanos, N. Kranitis, G. Theodorou, and A. Paschalis, "A 3.3 Gbps CCSDS 123.0-b-1 multispectral & hyperspectral image compression hardware accelerator on a space-grade SRAM FPGA," *IEEE Trans. Emerg. Topics Comput.*, vol. 9, no. 1, pp. 90–103, 2018, doi: 10.1109/TETC.2018.2854412.

[95] A. Tsigkanos, N. Kranitis, and A. Paschalis, "CCSDS 123.0-b-1 multispectral & hyperspectral image compression implementation on a next-generation space-grade SRAM FPGA," in *Proc. 6th ESA Int. Workshop Board Payload Data Compress*, 2018, pp. 21–22.

[96] J. Fjeldtvedt, M. Orlandić, and T. A. Johansen, "An efficient real-time FPGA implementation of the CCSDS-123 compression standard for hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 10, pp. 3841–3852, 2018, doi: 10.1109/JSTARS.2018.2869697.

[97] D. Báscones, C. González, and D. Mozos, "Parallel implementation of the CCSDS 1.2. 3 standard for hyperspectral lossless compression," *Remote Sens.*, vol. 9, no. 10, p. 973, 2017, doi: 10.3390/rs9100973.

[98] J. Nickolls and W. J. Dally, "The GPU computing era," *IEEE Micro*, vol. 30, no. 2, pp. 56–69, 2010, doi: 10.1109/MM.2010.41.

[99] A. Munshi, "The openCL specification," in *Proc. 2009 IEEE Hot Chips 21 Symp. (HCS)*, pp. 1–314, doi: 10.1109/HOTCHIPS.2009.7478342.

[100] C. A. Lee, S. D. Gasster, A. Plaza, C.-I. Chang, and B. Huang, "Recent developments in high performance computing for remote sensing: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 508–527, 2011, doi: 10.1109/JSTARS.2011.2162643.

[101] Y. Ma, L. Chen, P. Liu, and K. Lu, "Parallel programing templates for remote sensing image processing on GPU architectures: Design and implementation," *Computing*, vol. 98, nos. 1–2, pp. 7–33, 2016, doi: 10.1007/s00607-014-0392-y.

[102] L. Riha, J. Le Moigne, and T. El-Ghazawi, "Optimization of selected remote sensing algorithms for many-core architectures," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 12, pp. 5576–5587, 2016, doi: 10.1109/JSTARS.2016.2558492.

[103] J. E. Ball, D. T. Anderson, and C. S. Chan, "Comprehensive survey of deep learning in remote sensing: Theories, tools, and challenges for the community," *J. Appl. Remote Sens.*, vol. 11, no. 4, p. 042609, 2017, doi: 10.1117/1.JRS.11.042609.

[104] Á. Ordóñez, F. Argüello, D. B. Heras, and B. Demir, "GPU-accelerated registration of hyperspectral images using KAZE features," *J. Supercomput.*, vol. 76, no. 12, pp. 9478–9492, 2020, doi: 10.1007/s11227-020-03214-0.

[105] P. Quesada-Barriuso, D. B. Heras, and F. Argüello, "GPU accelerated waterpixel algorithm for superpixel segmentation of hyperspectral images," *J. Supercomput.*, vol. 77, no. 9, pp. 1–13, 2021, doi: 10.1007/s11227-021-03666-y.

[106] J. López-Fandiño, D. B. Heras, F. Argüello, and M. Dalla Mura, "GPU framework for change detection in multitemporal hyperspectral images," *Int. J. Parallel Program.*, vol. 47, no. 2, pp. 272–292, 2019, doi: 10.1007/s10766-017-0547-5.

[107] A. S. Garea, D. B. Heras, and F. Argüello, "Caffe CNN-based classification of hyperspectral images on GPU," *J. Supercomput.*, vol. 75, no. 3, pp. 1065–1077, 2019, doi: 10.1007/s11227-018-2300-2.

[108] M. Abadi et al., "Tensorflow: A system for large-scale machine learning," in *Proc. 12th {USENIX} Symp. Oper. Syst. Des. Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[109] S. Mittal, "A survey on optimized implementation of deep learning models on the Nvidia jetson platform," *J. Syst. Archit.*, vol. 97, pp. 428–442, Aug. 2019, doi: 10.1016/j.sysarc.2019.01.011.

[110] S. Zhang, Y. Su, X. Xu, J. Li, C. Deng, and A. Plaza, "Recent advances in hyperspectral unmixing using sparse techniques and deep learning," in *Hyperspectral Image Analysis*, S. Prasad and J. Chanussot, Eds. Cham: Springer-Verlag, 2020, pp. 377–405.

[111] W. Liu, L. Ma, and H. Chen, "Arbitrary-oriented ship detection framework in optical remote-sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 6, pp. 937–941, 2018, doi: 10.1109/LGRS.2018.2813094.

[112] X. Yu, X. Wu, C. Luo, and P. Ren, "Deep learning in remote sensing scene classification: A data augmentation enhanced convolutional neural network framework," *GISci. Remote Sens.*, vol. 54, no. 5, pp. 741–758, 2017, doi: 10.1080/15481603.2017.1323377.

[113] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, 2018, doi: 10.1109/TGRS.2017.2783902.

[114] Y. Chen, Y. Wang, Y. Gu, X. He, P. Ghamisi, and X. Jia, "Deep learning ensemble for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1882–1897, 2019, doi: 10.1109/JSTARS.2019.2915259.

[115] Á. Acción, F. Argüello, and D. B. Heras, "Dual-window superpixel data augmentation for hyperspectral image classification," *Appl. Sci.*, vol. 10, no. 24, p. 8833, 2020, doi: 10.3390/app10248833.

[116] J. M. Haut, A. Alcolea, M. E. Paoletti, J. Plaza, J. Resano, and A. Plaza, "GPU-friendly neural networks for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Sep. 2020, doi: 10.1109/LGRS.2020.3019378.

[117] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, 2016, doi: 10.1109/JIOT.2016.2579198.

[118] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 2017, doi: 10.1109/COMST.2017.2745201.

[119] L. Sun, L. Wan, and X. Wang, "Learning-based resource allocation strategy for industrial iot in UAV-enabled MEC systems," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 5031–5040, 2021, doi: 10.1109/TII.2020.3024170.

[120] Z. Rui, Z. Jingran, and W. Wukui, "Remote sensing imaging analysis and ubiquitous cloud-based mobile edge computing based intelligent forecast of forest tourism demand," *Distrib. Parallel Databases*, pp. 1–22, Jun. 2021, doi: 10.1007/s10619-021-07343-0.

[121] C. Cheng et al., "Adoption of image surface parameters under moving edge computing in the construction of mountain fire warning method," *PLoS One*, vol. 15, no. 5, pp. 1–16, May 2020, doi: 10.1371/journal.pone.0232433.

[122] R. P. Feynman, "Simulating physics with computers," *Int. J. Theor. Phys.*, vol. 21, nos. 6–7, pp. 467–488, Jun. 1982, doi: 10.1007/BF02650179.

[123] D. Deutsch and R. Penrose, "Quantum theory, the Church-Turing principle and the universal quantum computer," *Proc. Roy. Soc. London. A. Math. Phys. Sci.*, vol. 400, no. 1818, pp. 97–117, 1985.

[124] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proc. Roy. Soc. London. Ser. A, Math. Phys. Sci.*, vol. 439, no. 1907, pp. 553–558, 1992, doi: 10.1098/rspa.1992.0167.

[125] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Found. Comput. Sci.*, 1994, pp. 124–134, doi: 10.1109/SFCS.1994.365700.

[126] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, 1999, doi: 10.1137/S0036144598347011.

[127] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th Annu. ACM Symp. Theory Comput.*,

Association for Computing Machinery, 1996, pp. 212–219, doi: 10.1145/237814.237866.
[128] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Phys. Rev. Lett.*, vol. 103, no. 15, p. 150,502, 2009, doi: 10.1103/PhysRevLett.103.150502.
[129] M. A. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[130] A. M. Steane, "Simple quantum error-correcting codes," *Phys. Rev. A*, vol. 54, no. 6, p. 4741, 1996, doi: 10.1103/PhysRevA.54.4741.
[131] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018, doi: 10.22331/q-2018-08-06-79.
[132] M. Cerezo et al., "Variational quantum algorithms," *Nature Rev. Phys.*, vol. 3, no. 9, pp. 1–20, 2021, doi: 10.1038/s42254-021-00348-9.
[133] T. Kadowaki and H. Nishimori, "Quantum annealing in the transverse Ising model," *Phys. Rev. E*, vol. 58, no. 5, p. 5355, 1998, doi: 10.1103/PhysRevE.58.5355.
[134] A. Lucas, "Ising formulations of many NP problems," *Frontiers Phys.*, vol. 2, p. 5, Feb. 2014, doi: 10.3389/fphy.2014.00005.
[135] F. Arute et al., "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019, doi: 10.1038/s41586-019-1666-5.
[136] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017, doi: 10.1038/nature23474.
[137] V. Dunjko and P. Wittek, "A non-review of quantum machine learning: Trends and explorations," *Quantum Views*, vol. 4, p. 32, Mar. 2020, doi: 10.22331/qv-2020-03-17-32.
[138] A. Delilbasic, G. Cavallaro, M. Willsch, F. Melgani, M. Riedel, and K. Michielsen, "Quantum support vector machine algorithms for remote sensing data classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2021, pp. 2608–2611, doi: 10.1109/IGARSS47720.2021.9554802.
[139] P. Gawron and S. Lewiński, "Multi-spectral image classification with quantum neural network," in *Proc. 2020 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 3513–3516, doi: 10.1109/IGARSS39084.2020.9323065.
[140] D. A. Zaidenberg, A. Sebastianelli, D. Spiller, B. L. Saux, and S. L. Ullo, "Advantages and bottlenecks of quantum machine learning for remote sensing," in *Proc. 2021 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 5680–5683, doi: 10.1109/IGARSS47720.2021.9553133.
[141] S. Otgonbaatar and M. Datcu, "Classification of remote sensing images with parameterized quantum gates," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Sep. 2021, doi: 10.1109/LGRS.2021.3108014.
[142] S. Otgonbaatar and M. Datcu, "A quantum annealer for subset feature selection and the classification of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7057–7065, Jul. 2021, doi: 10.1109/JSTARS.2021.3095377.
[143] S. Otgonbaatar and M. Datcu, "Natural embedding of the Stokes parameters of polarimetric synthetic aperture radar images in a gate-based quantum computer," *IEEE Trans. Geosci. Remote Sens.*, early access, 2021, doi: 10.1109/TGRS.2021.3110056.
[144] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Satoshi Nakamoto Institute, White Paper, Oct. 2008. [Online]. Available: https://nakamotoinstitute.org/bitcoin/
[145] V. Buterin, "Ethereum whitepaper," White Paper, Nov. 2013. [Online]. Available: http://kryptosvet.eu/wp-content/uploads/2021/05/ethereum-whitepaper-kryptosvet.eu_.pdf
[146] "Blockchain and earth observation," European Space Agency, Paris, France, White Paper, Apr. 2019. [Online]. Available: https://eo4society.esa.int/wp-content/uploads/2019/04/Blockchain-and-Earth-Observation_White-Paper-April-2019.pdf
[147] "The land registry in the blockchain," Lantmäteriet, Telia Company, ChromaWay, and Kairos Future, White Paper, Jul. 2016. [Online]. Available: http://ica-it.org/pdf/Blockchain_Landregistry_Report.pdf
[148] M. Coca, I. Neagoe, and M. Datcu, "Physically meaningful dictionaries for EO crowdsourcing: A ML for blockchain architecture," in *Proc. 2020 IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 3688–3691, doi: 10.1109/IGARSS39084.2020.9324361.
[149] J. Benet, "Ipfs - Content addressed, versioned, P2P file system (DRAFT 3)," GitHub, 2015. https://github.com/ipfs/papers/raw/master/ipfs-cap2pfs/ipfs-p2p-file-system.pdf
[150] "FOAM: The consensus driven map of the world," Foamspace Corp., Sacramento, CA, USA, White Paper, Jan. 2018. [Online]. Available: https://www.foam.space/publicAssets/FOAM_Whitepaper.pdf
[151] "Community-based space platform," SpaceChain, White Paper, Mar. 2020. [Online]. Available: https://spacechain.com/wp-content/uploads/2020/01/spacechain-tech-whitepaper-271219.pdf
[152] J. W. Kocsis, "Researcher and NASA work to help spacecraft avoid floating debris," Univ. Akron, Akron, OH, USA, 2018. [Online]. Available: https://www.uakron.edu/im/news/researcher-and-nasa-work-to-help-spacecraft-avoid-floating-debris/
[153] U. Kalabić, A. Weiss, and M. Chiu, "Distributed small sat location verification," in *Proc. 2021 Integr. Commun. Navigation Surveillance Conf.*, pp. 1–12, doi: 10.1109/ICNS52807.2021.9441603.
[154] U. Kalabić, A. Weiss, and M. Chiu, "Orbit verification of small sat constellations," in *Proc. 2021 IEEE Int. Conf. Blockchain Cryptocurrency*, pp. 1–5, doi: 10.1109/ICBC51069.2021.9461065.
[155] S. Moeniralam, "A blockchain based data production traceability system," 2018. [Online]. Available: https://www.semanticscholar.org/paper/A-Blockchain-based-Data-Production-Traceability-Moeniralam/acd8fa5400a7b6f0e01c44a0e76d29f607921a27
[156] M. J. Molesky, E. A. Cameron, J. Jones, M. Esposito, L. Cohen, and C. Beauregard, "Blockchain network for space object location gathering," in *Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, 2018, pp. 1226–1232, doi: 10.1109/IEMCON.2018.8614769.
[157] M. Pincheira, E. Donini, R. Giaffreda, and M. Vecchio, "A blockchain-based approach to enable remote sensing trusted data," in *Proc. 2020 IEEE Latin Amer. GRSS ISPRS Remote Sens. Conf. (LAGIRS)*, pp. 652–657, doi: 10.1109/LAGIRS48042.2020.9165589.

GRS