Cloud Computing and Data Science
http://ojs.wiserpub.com/index.php/CCDS/

UNIVERSAL WISER PUBLISHER

Research Article

Exploring the Advancements in High-Performance Computing Paradigm for Remote Sensing Big Data Analytics

Sudha SK*, Aji S

Research Center, Department of Computer Science, University of Kerala, Thiruvananthapuram-695581, Kerala, India
Email: sudha.krishnaa@keralauniversity.ac.in

Received: 2 August 2023; **Revised:** 31 August 2023; **Accepted:** 7 September 2023

Abstract: The incredible growth in Remote Sensing (RS) data volume, with high spectral-spatial-temporal resolutions, has been utilized in various application domains. With the rapid advancements in modern sensor technologies, including 3D acquisition sensors, RS data with a large variety, velocity, veracity, varied value, and incredible volume are generated, leading to the Remote Sensing Big Data (RSBD). With the high availability of RSBD, we require High-Performance Computing (HPC) environments for storing and processing these High-Dimensional (HD), complex, heterogeneous, and distributed data. Also, introducing Deep Learning (DL) techniques in the RS domain demands more computing power, higher memory and networking bandwidth throughput capabilities, and optimized software and libraries to deliver the required performance. Motivated by this, we explore HPC computing environments for handling RSBD across multiple application domains in this paper. With particular emphasis on architectures such as cloud-based HPC, clusters, heterogeneous networks of computers, and specialized hardware architectures like Field Programmable Gate Arrays (FPGAs) and Graphics Processing Units (GPUs), we investigate how HPC technologies are being used to process RSBD efficiently while including integrated intelligence. This critical analysis results in a multi-layered cloud-based framework for efficient RSBD processing tasks. Also, we identified several data challenges to be handled while designing HPC frameworks. The findings from the study can help researchers better understand the HPC design concepts for developing RSBD frameworks.

***Keywords*:** remote sensing big data, high-performance computing, cloud computing, GeoAI, artificial intelligence

1. Introduction

As modern technologies, including 3D acquisition sensors and the Internet of Things (IoT), evolve, the size and volume of data that organizations have to work with are growing exponentially, demanding high computation powers to solve challenging problems in a timely fashion in various application domains. High-Performance Computing (HPC) refers to processing complex calculations at higher speeds across multiple servers in parallel using massive data volumes. Since HPC can take advantage of high volumes of data, it is becoming increasingly popular in various application domains, including Remote Sensing (RS). With high-resolution satellite missions, the enormously expanding volume of RS data received by Satellite Data Centers (SDC) increased by considerable Terabytes (TB) per day. With this, we have moved into the era of big Earth data called Remote Sensing Big Data (RSBD) [1]. Although RSBD provides a wide spectrum of real-world applications like disaster management, global security, land-cover mapping,

DOI: https://doi.org/10.37256/ccds.5120243480

climate and environmental studies, detecting forest fires, oil spill detection, precision agriculture and so forth, RSBD inevitably poses several additional challenges in processing. Three features mainly characterize big data: volume-variety-velocity, defined as three "V" dimensions [2]. For RSBD, the 3Vs could be extended due to the multi-source, multi-scale, dynamic-state, and non-linear characteristics. Thus, the RSBD, described by its dimensions like large variety, velocity, veracity, varied value and incredible volume (5Vs), made the analysis using traditional approaches more difficult for many applications, especially those involving real-time processing.

Chi et al. [3] addressed the understanding of three facets of big data from different perspectives related to i) who owns big data, ii) who have innovative big data methods and methodologies, and iii) who needs big data-based applications. With the huge availability of RS data with high spatial-spectral-temporal resolutions, conventional computing methods struggle to handle the new challenges of problem complexity, including data and model complexity [4]. In addition to the basic characteristics of big data, RSBD shows some specific features as below:

• Non-repeatability-observations of Earth objects and processes are unique in space and time and generally are not repeated.

• Uncertainty-occurs from indirect observation, sampling, and various recording techniques used in RSBD.

• Multi-dimensionality-results from a variety of data sources and sophisticated analytical techniques.

These characteristics result in high computational complexity in data analysis. Hence, understanding the computational advancement for handling the complexities and uncertainties in RSBD analysis is essential. With these constraints, using HPC frameworks for RS applications has become more widespread recently. Despite efficient feature selection and Dimension Reduction (DR) techniques, processing an extensive volume of multi-dimensional RSBD incurs extraordinary computational requirements. The HPC makes the massive, high-dimensional data loading, memory residing, and data transmission among processing nodes efficient.

The RS datasets are multi-dimensional and complex-structured metadata, making the standard Parallel-type File Systems (PFS) with stereo-typical physical data layouts no longer applicable. Also, the increasing necessity for real-time or near-real-time processing competence by several time-critical applications causes data-intensive issues to be substandard [5]. Moreover, as HPC systems evolve with time, so does their demand in various application domains. In this regard, knowing HPC concepts is essential to describe and understand the RSBD for specific applications. Developing computationally efficient techniques for transforming the massive volume of remote sensing data into scientific understanding is critical for many use cases. Several research efforts have recently been motivated towards incorporating HPC techniques and practices into remote sensing missions to address the abovementioned needs. This paper explicitly portrays the recent advancements in HPC models and how they are applied or introduced to RS problems and how AI is integrated into the HPC platforms. The study covers developments in various HPC architectures, like clusters, grids, clouds and specialized hardware components. The main contributions are listed as follows.

i) An extensive analysis of how the High-Performance Computing Paradigm (HPC-P) is introduced to enable various Large-Scale (LS) RSBD applications is presented.

ii) The challenges involved in designing and utilizing the HPC intelligently for RSBD are analyzed, and a new multi-layered cloud-based framework for RSBD processing is proposed.

iii) This paper provides a complete reference to essential concepts in HPC for RSBD applications and discusses how to integrate intelligence into the HPC-P.

The remaining sections of the paper are arranged as follows. **Section 2** briefly explains the detailed background study of the HPC-P concepts for RSBD. **Section 3** incorporates a compendium of algorithms and techniques used in the HPC-based RS data processing and how to integrate intelligence into HPC-P. Also, with the analyzed inferences, a new multi-layered cloud-based framework for RSBD-related tasks is proposed and discussed. **Section 4** presents the findings from the study and the future research challenges in designing HPC-based systems for RSBD applications with discussion, and **Section 5** concludes.

2. High-performance computing paradigm for RSBD analysis

Utilizing HPC systems for RS applications has become more widespread recently. Naturally, the RSBD acquired by various data centers are geographically distributed, and the HD characteristic complicates the distributed storing and accessing. To meet the computational requirements of extreme time-critical applications, researchers have begun

incorporating high-performance computing models in RS missions [6]. These HPC-based strategies become the most effective means of addressing the significant computing demands imposed by massive volume RS data. Several available high-performance platforms are actively employed to make sense of these RSBD processing tasks. The primary choices of computing platforms concentrate on cluster-based, supercomputers, or Cloud-based HPC systems [1]. The paper discusses several perspectives on utilizing the HPC paradigm for RSBD applications.

2.1 *Basic concepts of HPC-P in RS domain*

With the introduction of open-source approaches, HPC infrastructure has become more accessible. Hence, it is essential to understand the basic concepts of HPC-P to be applied to a specific domain. This will help the researchers in the RS domain to adapt to the new computing environments easily. Also, accelerated processing of RSBD requires strong computing power and faster computing speed, which becomes a challenging issue. HPC technology uses clusters of powerful processors operating parallelly to process massively multi-dimensional RSBD to resolve complex problems quickly. HPC systems often perform at incredible speeds than the fastest desktops, laptops or server nodes. Contrary to a standard computing platform, the HPC leverages the following:

• Massive Parallel Computing Environment: Run numerous processes concurrently on multiple servers, processors, or cores.

• HPC Clusters: include several networked high-speed computers, and a central scheduler manages the workload for parallel computing. High-performance Multi-Core Central Processing Units (MC-CPU) or Graphics Processing Units (GPU) are used, which are referred to as nodes, and are ideal for demanding data-intensive jobs, machine learning (ML) models, and calculations involving complex mathematics.

• High-Performance Components: include computing resources like memory, storage, distributed networking, and file systems, which are all high-speed, throughput, and low-latency devices that can enhance the cluster's processing speed and efficiency.

2.1.1 *HPC workload*

An HPC workload is a complex, data-intensive task spread across computing resources, running in parallel to balance the load of the processors in the heterogeneous environment. Depending on the level of interactions between the parallel processes operating concurrently, the workloads are divided into Loosely Coupled Workloads (LCW) and Tightly Coupled Workloads (TCW). In LCW, the multiple or parallel processes do not strongly interact with one another during the simulation. In contrast, in TCW, there is a frequent exchange of information between the parallel processes at each iteration or step of the simulation. Many architectures that apply to both LCW and TCW may require slight modifications based on the scenario, like i) traditional cluster environment, ii) batch-based architecture, iii) queue-based architecture, iv) hybrid deployment and v) serverless.

Each workload in the HPC is unique and uses a variable amount of CPU and memory to accomplish its tasks. The effort required depends on the task's duration, iterations, and scale. A workload essentially collects an input (I) and generates an output (O). Figure 1 displays the essential elements that make up the HPC workload.

The essential elements of an HPC workload consist of:

i) Request-refers to what is being requested, incorporating a set of read/write operations and the consecutive payload to/from a storage system.

ii) Application and Virtual Machines (VM)-Each workload is related to the tools or processes that are being employed in an application's ongoing task. The essential characteristics of a workload depend on how the application handles specific data and several inherent technical constraints.

iii) Working Set-is the amount of data generated or consumed throughout a workload. The data utilized in an average HPC workload is usually unstructured.

iv) Duty Cycle-is a sequence of events that occur, repeated, several times. The estimated repeatability of the work depends significantly on the application's goal, the user who is consuming data, and storage performance.

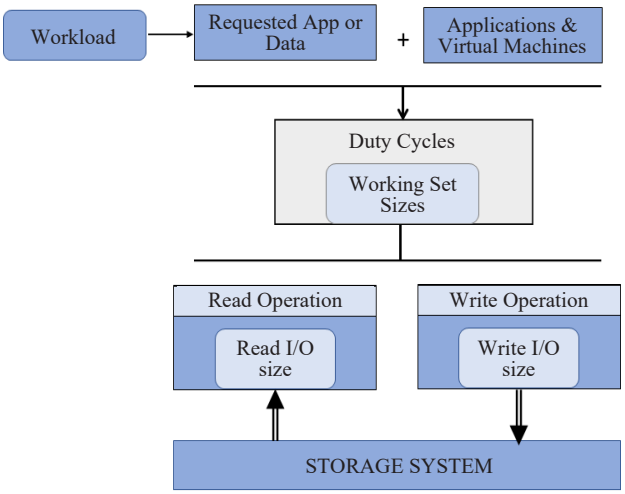

Figure 1. Components of a Workload in an HPC Environment

2.1.2 *Specialized hardware architectures for HPC-P*

The HPC environment needs specialized hardware to accelerate on-board satellite sensor platform processing. The on-board processing techniques reduce the cost and complexity of ground-processing systems, making them more affordable to a larger community. Specifically, this section focuses on two specialized hardware architectures used in HPC platforms: GPUs and FPGAs. Recently, GPUs have evolved into highly parallel, multi-threaded, multi-core processors with exceptional computational speed and incredible memory bandwidth [7]. The combined features like the supercomputing facility, high degree parallelism, high bandwidth memory, compact size, and low cost make systems integrated with GPU an appealing substitute to a massively parallelized system comprising commodity CPUs. The arrival of Nvidia's Compute Unified Device Architecture (CUDA), offering excellent programming capabilities of GPUs in a General-Purpose (GP-GPU) fashion, has introduced the feasibility of incorporating GPUs in plentiful RS applications. The powerful ability of GPUs attracted more researchers to make use of it as a cost-effective, high-performance computing platform, including those in the RS domain.

Reconfigurable computing, often known as FPGA-based computing, has recently gained popularity for implementing algorithms suited for RS applications [8-9] for accelerated performance. FPGAs exemplify an evolution over the Application-Specific Integrated Circuits (ASICs) [10] in the sense that they are explicitly designed to solve distinct problems. However, unlike FPGAs, the ASIC circuit cannot be altered after fabrication. Subsequently, reconfigurable hardware brings a trade-off between traditional hardware-software flexibility and performance by achieving hardware-like performance with software-like flexibility, which is significant in various RS applications [11]. The FPGAs can perform remarkably on-board using high-dimensional data in real time [12].

2.2 *Cloud-based HPC for RSBD*

HPC systems were previously constrained by the structural capability of on-premises infrastructures. Cloud computing platforms have recently added more resources to the local HPC capabilities. Cloud computing delivers a powerful and robust infrastructure for storing, accessing and analyzing datasets on very powerful servers, which virtualize supercomputers for user computations on a large scale. In a cloud environment, a group of virtualized dynamically scalable platforms and services are delivered on demand. Currently, cloud computing represents a cost-effective solution for data-intensive LS-RS applications. It offers data access transparency and elastic provisioning of resources in a 'pay-as-you-go' service model. So far, several cloud computing frameworks have been developed that support RSBD. A few important ones are i) Amazon Web Services (AWS), ii) Microsoft Azure and iii) Google Earth Engine (GEE). AWS and Azure serve as a 'pay-as-you-go' platform where users pay for the hours they use the services. Whereas GEE provides a major big geo-data processing platform that facilitates scientific discovery by providing free

accessibility to several RS datasets [13].

The main advantage of the GEE is the easy access to the RS data archives and HPC resources to process massive geospatial and RS datasets that are processed and periodically updated. GEE utilizes Google's computational infrastructure and available open-access RS datasets [14] and is accessed via an Application Programming Interface (API) and an Interactive Development Environment (IDE). GEE has several automatic parallel processing mechanisms and a fast computational platform to effectively deal with big data processing challenges. It also contains various built-in algorithms for planetary-scale data analysis, such as classification algorithms. It also helps scientists develop their own algorithms with less effort than before. More details on GEE can be obtained from [15]. AWS has a dedicated cloud-based Earth Observation (EO) offering called 'Earth on AWS' as part of its Public Dataset Program (PDP). It includes several open data from satellites like Landsat-8, Sentinel-1, Sentinel-2, China-Brazil Earth Resources Satellite (CBERS) program, National Oceanographic and Atmospheric Administration (NOAA) image datasets, and global model outputs with the largest suite of ML services. Azure contains Landsat and Sentinel-2 data from North America since 2013 and moderate-resolution imaging spectroradiometer (MODIS) imagery.

2.2.1 *Cloud deployment and service models*

The Cloud Deployment Model (CDM) defines a cloud environment and architecture, scalability of computing resources, accessibility to the services provided, etc. They also define relationships between the cloud infrastructure and the users. Different types of CDM are i) public cloud, ii) private cloud, iii) hybrid cloud, iv) community cloud, and v) multi-cloud. The various cloud service models provided for RSBD are i) Infrastructure as a Service (IaaS), ii) Platform as a Service (PaaS), and iii) Software as a Service (SaaS).

2.2.2 *Cloud-optimized data formats for RSBD*

Massive RSBD storage consists of raster data and metadata storage. RS raster data is generally stored as cloud-optimized data formats in an Object Storage System (OSS) or Distributed File System (DFS). In addition, it can be stored in NoSQL databases as tiles or arrays [16]. Cloud-optimized data storage formats for remote sensing data, such as Zarr and Cloud Optimized GeoTiff (COG) have emerged and improved the performance of RSBD data storage. RSBD metadata storage and management are mainly based on NoSQL, RDBMS, and NewSQL. Storage systems for RSBD based on MongoDB store both raster data and metadata to achieve integrated data/metadata storage [17-18].

2.2.3 *Cloud-based RSBD computing types*

The computation using RSBD in the cloud infrastructure can be grouped into i) data-separable computing and ii) data-inseparable computing. Data-separable computation is a series of independent subtasks by partitioning the datasets. It covers most RS analysis applications, such as pixel-based and tile-based analysis and has a simple parallelization strategy with better feasibility [16].

Data-inseparable computing cannot be parallelized by partitioning the data. There are dependencies between data-inseparable computing tasks, and thus, the input data should be homogeneous as data cubes or composite layers. The data-inseparable computing is mainly processed using MapReduce and array-based processing. Several studies have implemented various data-inseparable RS computations based on MapReduce-like technologies, such as K-Means clustering analysis [19], parallelized mosaics [20], pan-sharpening using Directed Acyclic Graph (DAG) [21], and object-based segmentation [22].

3. Algorithms and frameworks for HPC-based RSBD processing

This section includes an outlook on the algorithms, techniques and frameworks analyzed for this work. It is understood that the RSBD processing using an HPC environment uses multiprocessor systems like clusters and networks of computers, including massively parallel facilities. Low-level parallel programming models like Message Passing Interface (MPI), OpenMP and MPI + OpenMP are extensively utilized for RS image processing tasks. However, several works in the literature ensure that the HPC-based RS and geospatial data processing methods are considered efficient

with powerful algorithms. Implementing on-board processing algorithms for data reduction dramatically reduces data transmission rates [23]. Heterogeneous, distributed and parallel computing frameworks, in-memory computing and optimization methods are adopted in the HPC environment to empower the LS-RS data analysis. A review of cloud-based data management technology for RSBD storage and computing is presented [16].

An in-memory Spark-enabled distributed data mosaicking with geo-gridded data staging accelerated by Alluxio is explained [1]. A collaborative platform for offering data, algorithms, processing and analytic services to a number of users from different public and private user communities is provided [24]. A Convolutional Neural Network (CNN)-based deployment solution on resource-limited FPGA for spaceborne applications using RSBD is implemented [25]. A scalable computing resource model is developed to achieve fast processing of RSBD using a parallel distributed architecture [26]. A Spark-based adaptive real-time MapReduce data processing method that improves performance and stability is presented [27]. High-performance computational approaches enabling LS interferometric Synthetic Aperture Radar (SAR) processing are explained [28]. A multi-source remote sensing data integration framework based on a distributed management model using MongoDB cluster and spatial grid segmentation is presented [18].

A cloud-based model for processing massive RSBD with an optimized task scheduling scheme is proposed [22]. A new distributed architecture is proposed for the supervised classification of large volumes of EO data in a cloud computing environment [29]. Cloud-based HPC technique, which enables LS-RS data processing models as on-demand for several real-time services, is designed [30]. A real-time big data analytical architecture for RS satellite applications is proposed using Hadoop and MapReduce [31]. A cloud-based application, 'AgrCloud' provides services to process RS images using Hadoop and MapReduce [32]. However, the HPC requires a large degree of parallelism to make efficient use of the massive parallel computing power of the device. A lightweight cloud framework incorporating Spark-on-K8s is designed to improve the efficiency of a parallel RS image fusion algorithm [33] using the Elastic Computing Paradigm (ECP). A detailed description of the reviewed frameworks used for various types of RSBD tasks is listed in Table 1.

Table 1. Summary of Techniques Reviewed

Author	Technique/Framework	Task	Datasets
Ma Y. et al. [1]	Spark, Alluxio, Geotrellis	RS-Data Mosaicking	Spatial Resilient-Distributed Datasets
Cheng et al. [18]	MongoDB, Spatial-Grid Segmentation	RS-Data Distributed Storage Cluster	Gaofen-1, MODIS, OLI
Sun et al. [21]	MapReduce, Spark, vSphere	Pan-sharpening	QuickBird
Yan et al. [25]	FPGA, Parallel CNN	Spaceborne RS Applications	NWPU-RESISC45, DOTA
Guo et al. [26]	HDFS, Spark-on-Kubernetes, GeoPySpark	Assembling Big Data Storage, Parallel Computing, Real-Time Visualization	Resilient-Distributed Datasets
Tan et al. [27]	Spark, HDFS, MapReduce	RS Image Classification	Landsat
Quirita et al. [29]	Hadoop, MapReduce, PIG	RS-Data Classification	Pavia, Indian Pines
Wang et al. [30]	OpenStack, Hilbert-R+ Tree, GeoSOT	RS-Data Processing	Various Multi-Spectral, Multi-Temporal RS-Datasets
Rathore et al. [31]	BEAM VISAT, EnviView, Hadoop	RS-Data Analysis	ENVISAT-ASAR
P.Wang et al. [32]	Hadoop, MapReduce, HDFS	RS-Data Classification	Landsat, Spot, QuickBird
Huang et al. [33]	Spark-K8s, Elastic Computing, Containerized Hadoop	RS-Image Fusion	Sentinel 2A/B, PlanetScope

Many works use Spark-based parallel programming models to process massive RS images. The state-of-the-art (SOTA) engines adopt HDFS to store unprecedented volumes of RS big data. Also, data fusion algorithms are employed to deal with high spatial-temporal resolutions. Researchers focus more on scaling RSBD algorithms in diversified cloud deployments like public, private, hybrid, and community clouds. Moreover, we can see most Spark-based RSBD algorithms are implemented with low-level resilient distributed datasets (RDDs). From the investigation, some works indicate that containerized Spark algorithms are suitable for executing small tasks rather than big ones. Optimized task scheduling algorithms and auto-scaling cloud architectures to accelerate RS big data processing also ensure improved performances.

The 5V properties of the RSBD have high impacts on the data storage and distribution. The datasets mentioned in Table 1 are very complex in terms of 5Vs and their data management is carried out using efficient file structures, especially for time-critical analysis. Several works use the GEE platform to target the volume challenge of RSBD. The analysis has proven that intelligent and robust algorithms and techniques are utilized to dramatically improve the computing capability of HPC frameworks by carefully mapping the application processing entities on various processing units like CPUs and cores. Many of the HPC frameworks designed are efficient with the support of many topology mapping, optimization and list scheduling heuristic algorithms, which can reduce the complexities related to the 5Vs.

3.1 *Integrating intelligence into HPC-P*

To crunch the enormous amount of RSBD, we require smart Artificial Intelligence (AI) integrated into extremely powerful computing infrastructures. The HPC community has created several strategies to efficiently handle the difficulties of incorporating AI-at-scale, like i) including the necessity for more parallelism, ii) quicker I/O while using massive volume RS data sets, and iii) efficient traversing around a distributed computing environment. The computational intelligence in HPC relies on applying Expert-Level Heuristics (ELH) using DL inference to several processes, workloads, or simulations in unit time.

Cavallaro et al. [34] present advanced High-Performance and Disruptive Computing (HDC) technologies in the context of i) supercomputing and distributed computing, ii) specialized hardware computing, and iii) Quantum Computing (QC) for specialized Parallel Programming Models (PPM) and scalable algorithms as they play a significant role in the advancements of RSBD applications. Geospatial Artificial Intelligence (geoAI) is an emerging discipline that merges spatial science, AI, ML and DL with HPC to extract knowledge for data-intensive geospatial problems [35]. The intelligence can be integrated into the HPC environment through:

• FPGA and GPU support a high level of parallelization and help in accelerating the HPC and AI workload efficiency with low-weight and low-power integrated components that are essential to reduce payload and obtain results in realtime.

• Reconfiguring techniques and refining codes to operate more efficiently on HPC clusters with new optimization methods to accelerate data loading, pre-processing, training, and inference of workloads.

• Integrating intelligence into HPC requires developing AI-powered algorithms for RS applications with interfaces that support AI tools, such as Python and MATLAB.

• Since most AI-based HPC algorithms are bandwidth-hungry, HPC systems must move to a High-Bandwidth-Memory (HBM) configuration.

An extensive analysis carried out in the literature motivated us to propose a new multi-layered cloud computing framework, which is utilized for processing the massive volume RSBD for various application scenarios. The recommended framework is evaluated for specific RSBD tasks like RS segmentation, classification and retrieval.

3.2 *Multi-layer cloud-based framework for RSBD applications*

As remote sensing data plays an influential role in various human dimensions research, we recommend a new generalized representation of a multi-layer cloud-based framework for RSBD applications like segmentation, classification and retrieval with all the identified challenges and concepts. New data processing methodology and powerful computing architectures are essential for all RS application scenarios. Hence, the model presented in Figure 2 incorporates essential components for processing massive volumes of RS big data in a distributed fashion using

parallel implementation techniques on HPC machines containing several GPUs. Based on the specific RSBD application requirements, the model can be customized.

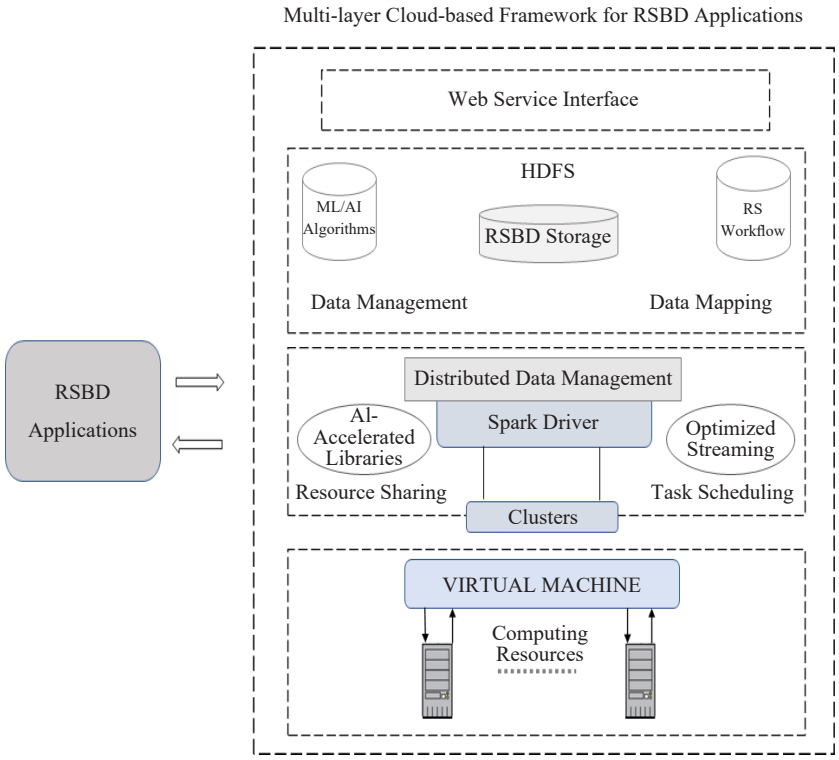

Figure 2. Generalized Representation of Multi-layer Cloud-based Framework for RSBD Applications

A multi-layered framework is preferred to figure out the issues related to various layers. The framework uses superior mapping algorithms to distribute massive RSBD. The AI-accelerated libraries and optimization provide more robust solutions for the RSBD applications of choice. For the proposed framework, highly scalable and parallel distributed architectures such as clusters or grids are used to train classifiers in reasonable time and provide users with a high-accuracy performance in the identified tasks. The new framework integrates a distributed data management and a task scheduling strategy into an optimization procedure to enable efficient and scalable processing of large-scale remotely sensed data.

The optimized streaming ensures reliable data access for data-intensive computing tasks. A Spark driver is preferred here as it can be used for batch processing, interactive queries, real-time stream processing, and graph computing. Also, it is a fast and open-source data-processing engine for ML and AI applications, backed by the largest open-source community in big data. Spark has a hierarchical master/slave architecture, with a *Spark Driver* as the master node that controls the cluster manager, which manages the worker (slave) nodes and delivers data results to the application client. The difficulty in distributed storing and accessing the high dimensional RSBD is resolved using efficient data mapping techniques.

4. Findings from the study

We derive a few findings from the critical analysis, and are discussed in this section. A performance comparison (in terms of Low, Medium and High) among various HPC computing platforms for RSBD processing is carried out and shown in Table 2.

Table 2. Performance Comparison of HPC Computing Platforms

HPC Using	RSBD Processing Capability	Ease of Use	Scalability	Reliability
Clusters	High	Low	Medium	Medium
Grid	High	Low	Medium	Medium
Cloud	High	Medium	High	High
GPU	Low	Medium	Low	Low
Multi-Core GPU	Low	High	Low	Low
FPGA	Low	Medium	Low	Low

The type of HPC should be selected based on the application scenarios and availability of other computing resources. The optimization strategies and libraries for ML and AI acceleration are to be used intelligently for improved performance. The future RSBD applications need integrated intelligence through reconfigurable techniques. The other key findings are:

• One distinction between an on-premises HPC system and one in the cloud is the ability to add and remove resources as needed via dynamic scaling.

• High-performance processing of the RS image manipulation algorithms is achieved through embedded parallelism.

• To achieve the highest feasible parallel I/O throughput, data-intensive RSBD applications require unique indexing algorithms and data placement strategies over the available disks.

• Heuristics-based scheduling algorithms are used to assign priorities to tasks and place them in a list ordered in decreasing magnitude of priority.

• Other than minimizing the execution time by exploiting the distributed and parallel computing capability of HPC platforms, some works insist on considering other objectives like minimizing energy consumption. These studies concentrate on the energy efficiency of GPUs and FPGAs, and more research is needed in the future.

4.1 *Future challenges to address in designing HPC for RSBD*

Designing unified frameworks and integrating intelligence with HPC will face various challenges in processing and handling RSBD due to the 5Vs. The HPC-P has empowered RSBD processing and makes it more possible than ever with extensive computing resources. Indeed, despite the benefits we could explore in cloud-based HPC-P, several obstacles remain to cloud adoption in the RS domain. A few challenges must be addressed as we look into the future demands of LS-RS applications. In these scenarios, efficient storing, managing, sharing and accessing of these distributed RS data at such an extreme volume and complexity is expected. Figure 3 shows some of the identified RSBD challenges to be unveiled by HPC-P.

Cloud-compatible applications facilitate RSBD processing and analysis, which can manage various ortho imagery, Digital Terrain Models (DEM), Point Cloud Data (PCD), etc. A significant challenge in designing HPC systems for RSBD is developing more heterogeneous systems that integrate resources from different locations. Besides the difficulties in handling the huge volume of RSBD, we have observed some other issues to be addressed while designing the HPC frameworks for various use cases in the RS domain. They are: developing effective programming models and languages for parallel computing, visualization tools, etc. Incorporating semantic and ontology-driven approaches to the HPC platform has to be explored more to understand, manipulate and analyze the RSBD. Another challenge in RSBD is how to deploy the data for real applications.

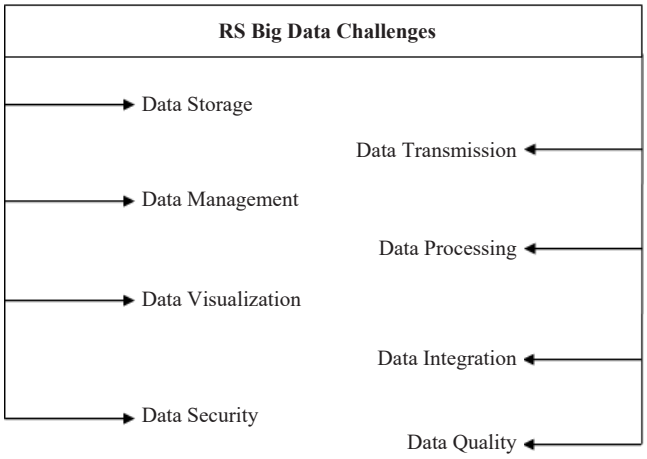

Figure 3. Identified RSBD Challenges

5. Conclusion

The high-performance computing paradigm recently got more recognition in RSBD applications as challenges posed by the increased amount of open data acquired daily by EO programs. The unique parallelized computing environments and programming techniques integrated into HPC systems could solve RSBD processing challenges for various use cases on a large scale. This paper explores bringing the HPC-P into the RS domain with integrated intelligence. The overall HPC concepts, technologies and software systems, and methods for integrating intelligence into the HPC-P with reconfigurable computing ability to resolve the problems involved in RSBD processing are discussed. As a contribution, this paper brings a complete reference for HPC design concepts from the perspective of RSBD. The critical analysis results in several findings and a new multi-layered cloud-based framework for RSBD tasks using HPC-P. Also, several future challenges are identified, including a requirement for real-time and on-demand RSBD processing using HPC models.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Ma Y, Wu H, Wang L, Huang B, Ranjan R, Zomaya A, et al. Remote sensing big data computing challenges and opportunities. *Future Generation Computer Systems*. 2015; 51: 47-60. Available from: doi:10.1016/j.future.2014.10.029.

[2] Laney D. *3D Data Management: Controlling Data Volume, Velocity, and Variety*. Application Delivery Strategies by META Group Inc; 2001. [Accessed 8th Sep 2023]. Available from: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf.

[3] Chi M, Plaza A, Benediktsson JA, Sun Z, Shen J, Zhu Y. Big-data for remote sensing: Challenges and opportunities. *Proceedings of the IEEE*. 2016; 104(11): 2207-2219. Available from: doi:10.1109/JPROC.2016.2598228.

[4] Zhong Y, Ma A, Ong YS, Zhu Z, Zhang L. Computational intelligence in optical remote sensing image processing. *Applied Soft Computing*. 2018; 64: 75-93. Available from: doi:10.1016/j.asoc.2017.11.045.

[5] Plaza AJ. Special issue: "Architectures and techniques for real-time processing of remotely sensed images." *Journal of Real-Time Image Processing*. 2009; 4(3): 191-193.

[6] Plaza AJ, Chang CI. *High-Performance Computing in Remote Sensing*. Chapman & Hall-CRC; 2007.

[7] Nickolls J, Dally WJ. The GPU computing era. *IEEE Micro Magazine*. 2010; 30: 56-69.

[8] Zhang X, Wei X, Sang Q, Chen H, Xie Y. An efficient FPGA-based implementation for quantized remote sensing image scene classification network. *MDPI Electronics*. 2020; 9: 1344. Available from: doi:10.3390/electronics9091344.
[9] Neris R, Rodríguez A, Guerra R, López S, Sarmiento R. FPGA-based implementation of a CNN architecture for the on-board processing of very high resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2022; 15: 3740-3750. Available from: doi:10.1109/JSTARS.2022.3169330.
[10] Buell DA, El-Ghazawi TA, Gaj K, Kindratenko VV. Guest editors' introduction: "High-performance reconfigurable computing." *IEEE Computer*. 2007; 40: 23-27.
[11] Thomas U, Rosenbaum D, Kurz F, Suri S, Reinartz P. A new software/hardware architecture for real time image processing of wide area airborne camera images. *Journal of Real-Time Image Processing*. 2009; 5: 229-244.
[12] Paz-Gallardo A, Plaza A. Clusters versus GPUs for parallel automatic target detection in remotely sensed hyperspectral images. *EURASIP Journal on Advances in Signal Processing*. 2010; 2010(1): 1-18. Available from: doi:10.1155/2010/915639
[13] Tamiminia H, Salehi B, Mahdianpari M, Quackenbush L, Adeli S, Brisco B. Google earth engine for geo-big data applications: A meta analysis and systematic review. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2020; 164: 152-170.
[14] Gorelick N, Hancher M, Dixon M, Ilyushchenko S, Thau D, Moore R. Google earth engine: Planetary scale geospatial analysis for everyone. *Remote Sensing of Environment*. 2017; 202: 18-27.
[15] Amani M, Ghorbanian A, Ali Ahmadi S, Kakooei M, Moghimi A, Mirmazloumi SM. Google earth engine cloud computing platform for remote sensing big data applications: A comprehensive review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2020; 13: 5326-5350. Available from: doi:10.1109/JSTARS.2020.3021052.
[16] Xu C, Du X, Fan X, Giuliani G, Hu Z, Wang W, et al. Cloud-based storage and computing for remote sensing big data: A technical review. *International Journal of Digital Earth*. 2022; 15(1): 1417-1445. Available from: doi:10.1080/17538947.2022.2115567.
[17] Wang S, Li G, Yao X, Zeng Y, Pang L, Zhang L. A distributed storage and access approach for massive remote sensing data in mongoDB. *ISPRS International Journal of Geo-Information*. 2019; 8(12): 533. Available from: doi:10.3390/ijgi8120533.
[18] Cheng Y, Zhou K, Wang J, Yan J. Big earth observation data integration in remote sensing based on a distributed spatial framework. *MDPI Remote Sensing*. 2020; 12(6): 972. Available from: doi:10.3390/rs12060972.
[19] Chebbi I, Boulila W, Farah IR. Improvement of satellite image classification: Approach based on hadoop/MapReduce. *2nd International Conference on Advanced Technologies for Signal and Image Processing*. IEEE; 2016. p.31-34. Available from: doi:10.1109/ATSIP.2016.7523046.
[20] Jing W, Huo S, Miao Q, Chen X. A model of parallel mosaicking for massive remote sensing images based on spark. *IEEE Access*. 2017; 5: 18229-18237. Available from: doi:10.1109/ACCESS.2017.2746098.
[21] Sun J, Zhang Y, Wu Z, Zhu Y, Yin X, Ding Z, et al. An efficient and scalable framework for processing remotely sensed big-data in cloud computing environments. *IEEE Transactions on Geoscience and Remote Sensing* 2019; 57(7): 4294-4308. Available from: doi:10.1109/TGRS.2018.2890513.
[22] Wang N, Chen F, Yu B, Qin Y. Segmentation of large-scale remotely sensed images on a Spark platform: A strategy for handling massive image tiles with the MapReduce model. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2020; 162: 137-147. Available from: doi:10.1016/j.isprsjprs.2020.02.012.
[23] Lee CA, Gasster SD, Plaza A, Chang C-I, Huang B. Recent developments in high-performance computing for remote sensing-A review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2011; 4(3): 508-527. doi: 10.1109/JSTARS.2011.2162643.
[24] Picchiani M, Maranesi M, Mastrucci M, Coltea IG, Pompei G, Di Giacomo L. The dydas-"Dynamic data analytics services" platform for HPC big data analytics of earth observation and geospatial data. *IGARSS 2022-IEEE International Geoscience and Remote Sensing Symposium*. Symposium, Kuala Lumpur, Malaysia; 2022. p.4011-4014. Available from: doi:10.1109/IGARSS46834.2022.9884375.
[25] Yan T, Zhang N, Li J, Liu W, Chen H. Automatic deployment of convolutional neural networks on FPGA for spaceborne remote sensing application. *MDPI Remote Sensing.* 2022; 14: 3130. Available from: doi:10.3390/rs14133130.
[26] Guo J, Huang C, Hou J. A scalable computing resources system for remote sensing big data processing using GeoPySpark based on Spark on K8s. *MDPI Remote Sensing*. 2022; 14: 521. Available from: doi:10.3390/rs14030521.

[27] Tan X, Di L, Zhong Y, Yao Y, Sun Z, Ali Y. Spark-based adaptive mapreduce data processing method for remote sensing imagery. *International Journal of Remote Sensing.* 2021; 42(1): 191-207. Available from: doi:10.1080/01431161.2020.1804087.

[28] Imperatore P, Pepe A, Sansosti E. High performance computing in satellite SAR interferometry: A critical perspective. *MDPI Remote Sensing*. 2021; 13(23): 4756. Available from: doi:10.3390/rs13234756.

[29] Quirita VAA, da Costa GVO, Happ PN, Feitosa RQ, Ferreira RS, Oliveira DAB, et al. A new cloud-computing architecture for the classification of remote sensing data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.* 2017; 10(2): 409-416. Available from: doi:10.1109/JSTARS.2016.2603120.

[30] Wang L, Ma Y, Yan J, Chang V, Zomaya AY. pipsCloud: High-performance cloud-computing for remote sensing big-data management and processing. *Future Generation Computer Systems*. 2016; 78(1): 353-368. Available from: doi:10.1016/j.future.2016.06.009.

[31] Rathore MMU, Paul A, Ahmad A, Chen B-W, Huang B, Ji W. Real-time big data analytical architecture for remote sensing application. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2015; 8(10): 4610-4621. Available from: doi:10.1109/JSTARS.2015.2424683.

[32] Wang P, Wang J, Chen Y, Ni G. Rapid processing of remote sensing images based on cloud-computing. *Future Generation Computer Systems*. 2013; 29(8): 1963-1968. Available from: doi:10.1016/j.future.2013.05.002.

[33] Huang W, Zhou J, Zhang D. On-the-fly fusion of remotely-sensed big data using an elastic computing paradigm with a containerized spark engine on kubernetes. *MDPI Sensors*. 2021; 21: 2971. Available from: doi:10.3390/s21092971.

[34] Cavallaro G, Heras DB, Wu Z, Maskey M, López S, Gawron P, et al. High-performance and disruptive computing in remote sensing: HDCRS-A new working group of the GRSS earth science informatics technical committee [Technical Committees]. *IEEE Geoscience and Remote Sensing Magazine.* 2022; 10(2): 329-345. Available from: doi:10.1109/MGRS.2022.3145478.

[35] Li W. GeoAI: Where machine learning and big data converge in GIScience. *Journal of Spatial Information Science*. 2020; 20: 71-77. Available from: doi:10.5311/JOSIS.2020.20.658.