

FPGA Accelerator for Meta-Recognition Anomaly Detection: Case of Burned Area Detection

Mihai Coca  and Mihai Datcu , *Fellow, IEEE*

Abstract—Optical remote sensing instruments accumulate abundant data from across all of the earth’s land surfaces, making it possible both to understand the effects of climate change and to monitor, investigate, and manage ground-level events in detail. Processing data using resources located near on-board satellite sensors can bring major benefits in terms of minimizing analysis time and quickly initiating active actions in critical situations. In satellite missions, long-term production on-board algorithms may encounter unexplored samples, i.e., abnormal ground-level events, and need to be able to discriminate and take the correct action. In this matter, the authors present a field programmable gate array (FPGA)-based solution for natural anomaly detection in multispectral imagery using deep convolutional neural networks. The effects of weather-induced hazards and natural disasters, considered anomalies in this sense, are discovered by modeling an anomaly detector on a hybrid system that is hardware efficient. The proposed approach is assembled on a Xilinx Zynq UltraScale+ XCZU9EG multiprocessor system-on-chip (MPSoC) device, where a deep convolutional model is scaled into the FPGA logic, followed by a downstream statistical meta-recognition predictor. The proposed anomaly detection accelerator has produced notable results in identifying a contemporary natural hazard, i.e., burned areas, in scenes acquired by Sentinel-2 over Europe, i.e., Spain and France. The implemented algorithm achieved on the FPGA accelerator an equivalent speedup of $4.46\times$ and $4.5\times$ lower power consumption than the equivalent implementation on the Tesla K80 GPU.

Index Terms—Anomaly detection, burned area detection, field programmable gate array (FPGA), on-board processing, remote sensing.

I. INTRODUCTION

ANOMALIES are rare patterns in data observations that deviate from a perception of normal and predicted behavior. Classical concepts for unsupervised anomaly detection, extended across multiple disciplines (e.g., statistics, medicine, engineering, natural sciences), consists of principal component analysis (PCA) [1], nearest neighbor algorithms [2], the one-class support vector machine (OCSVM) [3], and support vector data description (SVDD) [4], among others. Due to the

absence of labeled data to characterize the anomalousness in real scenarios, a dominant solution for detecting anomalies is to learn a model from normal data and try to deflect anomalies.

Anomaly detection techniques dealing with satellite imagery are either concerned with any changes in an image over time, i.e., change detection [5], or in areas that appear abnormal on the stationary image [6]. Frequently, anomalies are identified with man-made targets. However, the anomalies produced by natural disasters on the earth’s surface are of more interest due to their high-scale damaging potential, e.g., fire spot in a forest, oil spill in the sea [7].

Event detection in earth science is often critical for immediately addressing negative impacts on natural resources, e.g., drought-related vegetation disturbances [5], devastating floods [8], active fire detection [9]. Wildfire is the most extreme natural hazard that has caused serious damages to human safety and natural ecosystems in recent years, i.e., Australian areas affected by wildfire in 2019 [10], ravage outbreak of wildfires in Bolivia in 2019 [11], large fire events in California in 2020 [12]. This disastrous event has an annual cycle in predisposed places, but new places are constantly appearing around the world due to climate change. Accurate location at an early stage is of great importance in fire-fighting strategies. Global coverage characteristics of satellite imagery combined with computer vision mapping techniques represent one of the preferred alternatives for operational wildfire surveillance. High-resolution sensors placed on-board a satellite provide huge amounts of information that are impossible to transmit wholly and in real-time to the ground station. Automatic data processing near the sensor, immediately after acquisition, can reduce the information flow and promote early detection of anomalous events.

Space-based missions generate complex computing needs because limited resources must be used in harsh environments with big temperature ranges, radiation, vacuum and vibration. The environmental conditions as well as the limited power generation capabilities on-board restrict the hardware resources used and decrease the capability and complexity of the algorithms implemented on-board. Small-satellite (SmallSat) missions are low-cost platforms that provide programmatic flexibility, monitoring strategy in constellations, and distributed capacities, all composed in a scalable architecture [13]. Field-programmable gate arrays (FPGAs) are adaptive devices related to reconfigurable computing, highly desirable for space applications, e.g., image processing and image compression [14]. FPGAs empower the utilization of algorithmic parallelism in application-specific architectures and provide multiple high-

Manuscript received 12 December 2022; revised 7 March 2023; accepted 19 April 2023. Date of publication 5 May 2023; date of current version 14 June 2023. This work was supported by the Romanian Ministry of Education and Research, CNCS - UEFISCDI, under Project PN-III-P4-ID-PCE-2020-2120, within PNCDI III. (Corresponding authors: Mihai Coca; Mihai Datcu.)

Mihai Coca is with the Military Technical Academy “Ferdinand I”, 050141 Bucharest, Romania, and also with the Research Center for Spatial Information (CEOSpaceTech), University Politehnica Bucharest (UPB), 060042 Bucharest, Romania (e-mail: mihai.coca@mta.ro).

Mihai Datcu is with the Earth Observation Center (EOC), German Aerospace Center (DLR), 82234 Wessling, Germany, and also with the Research Center for Spatial Information (CEOSpaceTech), University Politehnica Bucharest (UPB), 060042 Bucharest, Romania (e-mail: mihai.datcu@dlr.de).

Digital Object Identifier 10.1109/JSTARS.2023.3273309

bandwidth input interfaces at a more energy-efficient cost than a general-purpose central processing unit (CPU). Hybrid architectures, e.g., system-on-chip (SoC) devices, pursue mixing diverse computing technologies in order to achieve cumulative improvements by balancing all incorporated benefits. SoC devices integrate certain computing architectures by combining general-purpose blocks, i.e., CPU, with specialized blocks, i.e., FPGA, graphic processing unit (GPU), digital signal processor (DSP). These attractive combinations in the architecture of a system make it easy to split algorithms into subroutines run on the most suitable execution blocks, all with performance gains. Xilinx's UltraScale architecture broadens FPGA capability for space applications, enabling high-throughput satellite services [15].

On-board processing for space applications is currently limited due to development and qualification cycles. Real-time data processing is achieved through hybrid processing systems such as SoC devices, which are essential for small-satellite missions. PhiSat-1 nanosatellite mission was the first earth observation satellite with AI on-board, using a CNN model to detect clouds in images and filter out unusable data to maximize relevant information downlinked to the ground segment [16], [17].

The challenges of processing data on embedded devices are significant and require innovative solutions. This article presents a novel anomaly detection architecture based on the concept of meta-recognition [18]. By incorporating a postrecognition classifier that makes predictions on the last activation layer of a trained DCNN model, the proposed anomaly detector can effectively discover anomalies in multispectral scenes. The proposed model is assembled as software for an integrated multiprocessor system-on-chip (MPSoC) device, suitable for small-satellite missions.

The main contributions of this work are briefly described as follows:

- 1) codesign of a hardware-software architecture for natural anomaly detection in multispectral imagery;
- 2) verification of the implemented hardware-software partitioned anomaly detection system on a Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit for two real multispectral datasets with scenes affected by forest fires.

The rest of this article is organized as follows. First, a comprehensive review of related works is provided in Section II. Sections III and IV introduce the reader to the proposed methodology and datasets used in evaluation stage. Results on the presented datasets are provided in Section V. Finally, Section VI concludes this article.

II. RELATED WORK

The traditional on-board real-time classification and detection methods for remote sensing (RS) imagery are almost entirely based on the computation of the data correlation matrix R or covariance matrix Σ and their inverses, with the aim to continuously update the inverses as pixels are being received [19]. For a known target to be detected d , common features are computed using R^{-1} for data whitening and matched filter d^T for detection or classification. In the case of anomaly detection when the required target x is unknown, the pixel itself x^T serves as the

term to be matched [20]. The most popular anomaly detection algorithm is Reed-Xiaoli (RX) [20] that computes a background covariance matrix and its inverse, followed by a distance measurement. This algorithm concentrates on detecting pixels whose spectral signatures are in contrast with their background by using Mahalanobis distance. Due to the parallel and distributed nature of RX algorithm, a recursive variant was deployed on FPGA fabric [21] for on-board processing convenience. The authors proposed a framework that used an off-chip memory for data caching and a processor for two tasks, updating the inverse matrix R^{-1} and filtering for anomaly detection.

Progressive line processing methods have shown a growing interest in anomaly detection applications under real-time constraints. In [22], [23], and [24], the authors proposed a line-by-line fast anomaly detector for hyperspectral imagery (LbL-FAD) that uses an orthogonal projection strategy to separate the background distribution. This hardware-friendly anomaly detector variant computes an orthogonal subspace on background samples in order to better detect anomalous pixels. In [23], the LbL-AD algorithm is evaluated on real images acquired by a push-broom sensor mounted on an unmanned aerial vehicle (UAV). Further, in [24], the LbL-AD algorithm proved an attractive tradeoff between time performance, energy consumption, and cost of processing a hyperspectral image of $825 \times 1024 \times 160$ pixels. Zhang et al. [25] suggested a real-time causal line-wise progressive anomaly detection (RCLP-AD) using fast Cholesky decomposition. Their work improved real-time computational performance and contributed to numerical stability in the background suppression process.

The current state of RS imaging processing technology increasingly includes AI components that demand more computational resources due to increased complexity. Recent advances in microelectronics offer hardware accelerators with an optimised computation-to-power ratio, allowing AI algorithms to be deployed close to the sensor. PhiSat-1 mission [16] proved the utility of AI as a robust and efficient method for accurate cloud detection on-board. This demonstration mission emphasized the robustness of the on-board Intel Movidius Myriad 2 hardware accelerator and the accuracy of the CloudScout convolutional neural network [17]. The CloudScout classifier achieved a 96% accuracy when performing cloud detection on live images on-satellite.

In real-time detection of anomalous targets from on-board multispectral imagery, the processing speed and the accuracy are equally important but difficult to achieve simultaneously. Ma et al. [26] proposed a pruning-quantization-anomaly-detector (P-Q-AD) that balances accuracy and throughput by shrinking a neural model by removing the redundant neurons. Lei et al. [27] proposed a fast spectral-spatial anomaly detection algorithm called Fast-MGD, which uses morphological reconstruction and a simplified guided filter to achieve high detection accuracy with simple filtering techniques. The algorithm utilizes a deeply pipelined acceleration scheme designed for hardware implementation on FPGAs that handle hyperspectral images.

Pixel-based anomaly detection algorithms, i.e., RX Detector [20], extract targets that are spectrally distinct from the image background with good performance. In order to be effective,

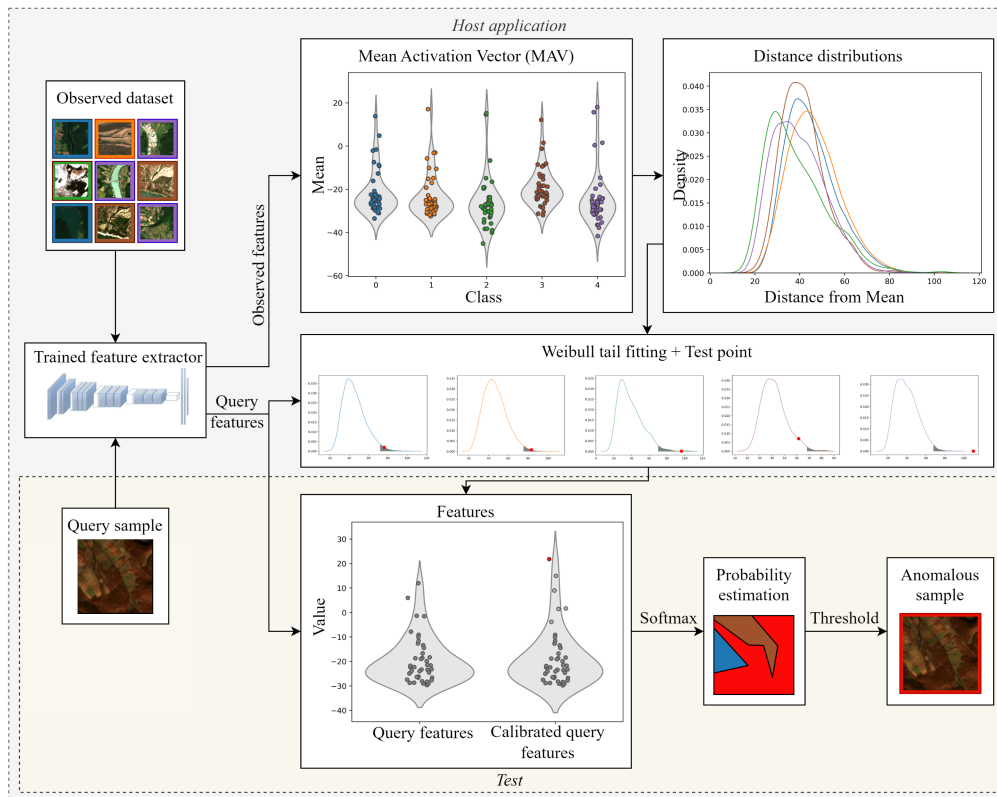


Fig. 1. Host application for anomaly detection. After training a deep convolutional neural network, the same training set, i.e., observed dataset, is used to obtain the features of correctly classified samples. The features are further used to compute an activation vector per class containing the mean of the activation vectors (MAV) of the samples belonging to that class. Next, a distance vector per class is calculated which contains the distances between all samples of that class and the previously calculated mean, i.e., distances between vectors. Subsequently, the tails of the distance distributions are modeled by Weibull distributions. In the testing phase, the characteristics of a query sample determine its position in the distance distributions of all classes encountered in the observed dataset. Depending on the result of the Weibull estimation on the test point, i.e., the query sample, the α top features of the test point are calibrated resulting in a new scalar value, i.e., the equivalent weight for the anomaly class, in the component of the initial feature vector. The new feature vector generates via a normalized exponential function a probability distribution. Finally, based on a threshold applied to the probability distribution, a binary decision is obtained, i.e., whether it is an anomaly or not.

the anomalous targets must be sufficiently narrow relative to the background and expected to follow a Gaussian distribution. However, in complex multispectral images with convoluted feature relations, the Gaussian assumption fails in providing an accurate model for the real data, particularly at the tails of the distribution. With this in mind, in the proposed method, open and broad patch-level anomalies were searched for in large heterogeneous multispectral scenes.

III. PROPOSED METHODOLOGY

In most situations, the trained classifiers are implemented under closed set conditions, since the classes encountered during implementation are known and are exactly the same as during training. In uncontrolled real-world environments, it is very likely to encounter instances of classes that have not been covered by the training data [28]. In remote sensing (RS) scenes, anomaly detection algorithms address a discovery task of small-scale portions that do not harmonize with the background. Considering a collection of observations from a number of normal classes, e.g., usual land cover classes, a query sample that does not belong to one of those current classes would be identified with a new abnormal class, e.g., burned areas.

Consequently, it was proposed an algorithm in which the deep latent values extracted from an observed dataset are extended to estimate the probability that a given input is associated with an abnormal class that was not seen in the training phase. In this context, it was defined as a framework that involves two consecutive stages, the first for implementing, training, and testing the algorithm on a GPU cluster, followed by an optimization and deployment phase of the algorithm on a low-resource platform. For the first stage, Fig. 1 depicts the proposed method as a host application that uses the deep latent characteristics of a test sample to determine if it belongs to the abnormal class. The abnormal class, a new class introduced in the nomenclature of training classes, characterizes all samples that deviate from the observed categories, and finally considers anomalies. For the second stage, Fig. 2 describes the flow used to build the proposed algorithm as an anomaly detection accelerator and deploy it on an embedded MPSoC device equipped with an FPGA device. In this stage, the *host application* block incorporates the algorithm described in Fig. 1.

The proposed anomaly detection algorithm contains three steps, i.e., data preprocessing, background modeling, and similarity measurement. In the preprocessing step, data are prepared by standardization [29] for use as input to a DCNN classifier.

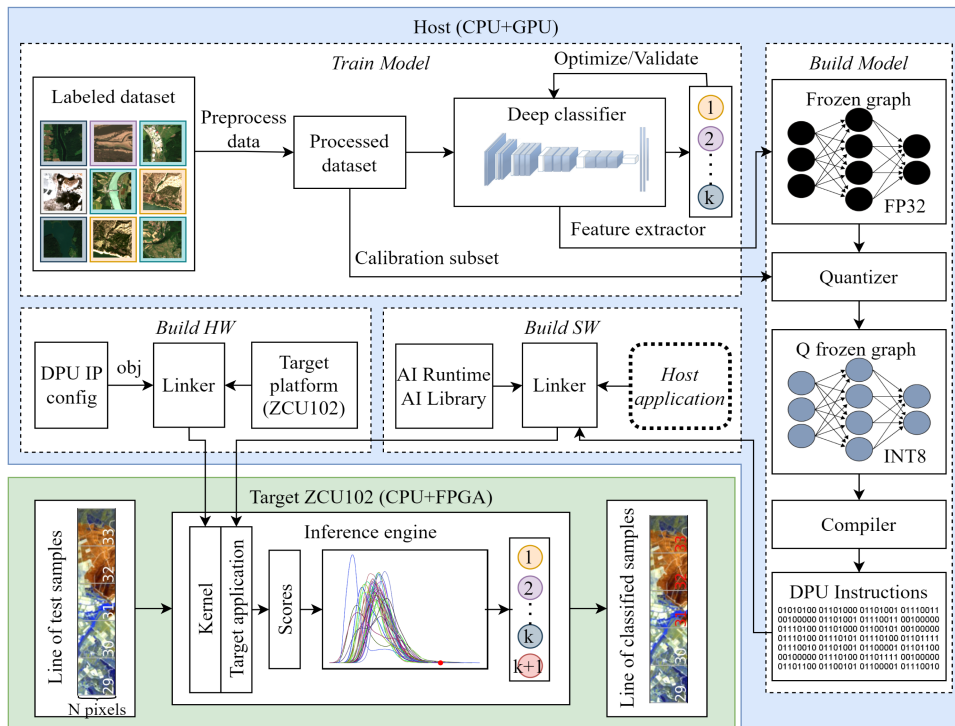


Fig. 2. General flowchart for the optimization and deployment of the on-board anomaly detection accelerator. Training a model on a host GPU device generates a feature extractor working with FP32 data. Quantifying the feature extraction graph in INT8 data and compiling it into a specific instruction set compress the model for the target platform, i.e., ZCU102. The host application described in Fig. 1 is integrated with the compiled feature extractor and a suite of AI libraries to get the target application running on the target platform. The hardware component running on the FPGA is obtained by configuring a DPU instance. On the embedded platform, the proposed algorithm detects anomalies from a set of test samples.

Afterward, the classifier is trained on observed data and used as a feature extractor, e.g., data reduction. In the second step, background statistical features are approximated through fitting Weibull distributions on each observed class in the training dataset (see Fig. 1). Finally, a distance metric is utilized for the similarity metric, e.g., Euclidean and Cosine distances, and a threshold is selected for a binary decision.

A difficult problem in implementing a DCNN on the FPGA is the long design time as well as the substantial effort involved. Xilinx, Inc. has released a group of parameterizable intellectual property cores called deep learning processing unit (DPU) [30], specialized in efficiently implementing various DCNN architectures on FPGA by supporting diverse deep learning operations. The DPU has a specialized instruction set utilized in the DL algorithm implementation. The acceleration development flow involved a unified software stack called Vitis AI Framework, provided by Xilinx, to convert, i.e., quantizing and compiling, a Tensorflow DCNN model into a compatible format supported by the DPU engine. Moreover, the Vitis software permitted concurrent development and test of both the software component (i.e., the target application), and the hardware component (i.e., the kernel), contained in a heterogeneous application.

In the optimization and deployment phase of the algorithm (see Fig. 2), a DCNN model learned with the Tensorflow framework was converted from a floating-point format to a single-precision format using a quantizer. In addition to the frozen graph of the DCNN model, the quantizer needed as input a

calibration subset of the observed dataset to evaluate the distribution of activations and to mitigate the accuracy degradation. Next, a compiler translated from Tensorflow specific format to DPU engine specific format. Next, the software build step, i.e., *Build SW*, generated for the target platform an inference-only model that was called by the *host application*. Thereafter, the target application effectively used the hardware-built DPU via a specific library, i.e., AI Library, and AI Runtime, resulting in an executable file for a specific FPGA target board.

The approach proposed in this article considers a methodology that adapts DCNNs for open set recognition [31]. This methodology introduced a new model layer, i.e., OpenMax, which estimates the probability of an input belonging to an unknown class. The OpenMax model is based on meta-recognition concepts adapted to the activation patterns in the penultimate layer of a classifier. The OpenMax tier is an extension of the last activation function, e.g., SoftMax, of a DCNN that normalizes the output of a network to a probability distribution over the initial predicted output classes plus an anomalous class. The resulted likelihood was used to estimate the probability of a given input belonging to an anomalous class. Further, the activation vectors from the penultimate layer were utilized to estimate if the test input is out of distribution from known training data.

A. Meta-Recognition

Meta-recognition [18] is a prediction method that uses statistical extreme value theory (EVT) for postrecognition score

analysis. In particular, after a system produces a series of distances or similarity scores for a particular test sample, a predictor will produce a decision of recognition success or failure. Distributional modeling and machine learning, among others, may be the basis for a postrecognition classifier. Scheirer et al. [18] proposed a statistical theory of postrecognition score analysis, i.e., *meta-recognition*, derived from the extreme value theory. They developed a statistical classifier based on the Weibull distribution and demonstrated that a statistical meta-recognition is substantially better than a standard threshold test over the original score data.

Multiclass classifiers in deep learning applications are usually trained on a closed-set dataset. In contrast to the closed set nature, recognition may be assimilated with the open set attribute of a real world classifier that rejects unseen classes at query time. A DCNN classifier applies on the last activation layer a normalized exponential function, i.e., SoftMax, which generates a probability distribution over a known and fixed number of class labels, i.e., k . At test time, this classifier trained only on the closed set samples may erroneously label with high certainty an anomalous sample belonging to one of the classes in the closed set. For an unseen input, it is unlikely to produce such low probabilities for all known classes that a threshold would reject the input and classify it as an anomaly. In practice, the deep network activations are not bounded, ensuring that SoftMax cannot regulate the open space recognition [32]. Starting from this hypothesis, the meta-recognition task was involved in this work to inspect the scores and identify when a DCNN classifier is possibly incorrect in evaluation.

Extreme value theory holds that the extreme values of a continuous distribution accept only three types of parametric forms, specifically Gumbel, Frechet, reversed Weibull distributions [18]. Minimum pairwise distance of a test sample to the closest sample in the closed set is proved through EVT that follows a Weibull distribution. The argumentation is that extreme features in contrast to average ones are the most essential for discriminating between different objects [33]. This facilitates the development of an inclusion map to decide if a test sample is in class with the minimal pairwise distance or is classified as an anomaly. In this article, building on the multiclass meta-recognition concept proposed in [31], the mean scores from the per-class estimation layer, i.e., logit layer, of a trained deep network classifier is used to predict if an input sample is separated from the known trained dataset. In the first stage, EVT is used to fit the postrecognition activations, i.e., mean activation vectors (MAVs) of the logit layer, of the known classes to Weibull distributions. For Weibull fitting, the FitHigh function is used, which is available in the libMR library [18].

Remote sensing images need a rich characterization through multilabeling since they contain a huge diversity of semantically complex content. Starting from this position, a convolutional classifier must use a sigmoid layer as the last activation layer in order to classify multilabel samples. A multilabel feature extractor is trained on normal classes C_i , $1 \leq i \leq k$ to compute for each class the corresponding MAV, $\mu_i = \text{mean}(v_{ij})$, $1 \leq i \leq k$, $1 \leq j \leq |C_i|$ using the activation vector v , i.e., the logit

values (see Fig. 1). With S_n was noted the set of indexes, in descending order, of the highest n probabilities generated by a final Sigmoid layer on activation vector v . Next, for each class C_i , a Weibull model $\rho_{C_i} = (\tau_{C_i}, \lambda_{C_i}, \kappa_{C_i})$, $1 \leq i \leq k$ with three parameters, data shifting τ , Weibull scale λ , respectively Weibull shape κ , is computed. Parameter τ is dynamic and depends on the data itself, i.e., is the smallest score (distance) on the tested activation vector v , aiming to shift v in zero (1a). Model $\rho_{C_i}(x)$ provides meta-recognition estimated probability that determines if query sample x is anomalous or not.

In the second phase, the activations of a query sample x adjust α top activations, i.e., α top probabilities given by a Sigmoid layer, by approximating the Weibull distribution function (1a)

$$w_i(x) = 1 - f(i) \left(1 - e^{-\left(\frac{x-\tau_i}{\lambda_i}\right)^{\kappa_i}} \right), \quad 1 \leq i \leq k \quad (1a)$$

$$f(i) = \begin{cases} \frac{\alpha + 1 - \text{rank}_i(S_\alpha(v))}{\alpha} & \text{if } i \in S_\alpha(v) \\ 0 & \text{otherwise} \end{cases}. \quad (1b)$$

After computing Weibull CDF on the distance between query sample x and MAVs of α top activations, revised activation vector is computed (2a), where operator \circ is used for the scalar product between two vector.

$$\hat{v}(x) = v(x) \circ w(x) \quad (2a)$$

$$\hat{v}_{k+1}(x) = \sum_{i=1}^k v_i(x)(1 - w_i(x)). \quad (2b)$$

Afterward, a pseudoactivation for the unseen class C_{k+1} is computed while preserving the total activation level constant (2b). Finally, the rejection is decided on the revised probabilities of normal classes C_i , $1 \leq i \leq k$, with regard to the anomalous class C_{k+1} (3)

$$p_i(x) = \frac{e^{\hat{v}_i(x)}}{\sum_{j=1}^{k+1} e^{\hat{v}_j(x)}}, \quad 1 \leq i \leq k+1 \quad (3)$$

where x is anomalous if p_{k+1} is the highest in p_i , $1 \leq i \leq k+1$ or higher than a threshold η .

B. Model Quantization and Compilation

Quantization scales down the bit width of the parameters and activation values processed in a DCNN. Smaller data types are appealing for energy efficiency and storage cost area because they produce a reduced model size with minor deterioration in accuracy [34]. Another reason for using fixed-point represented data is that the DSP unit in the FPGA is better at handling fixed-point numbers. In this work, the quantizer is used to convert numerical values of model weights from 32-b floating point to 8-b fixed point precision, reducing computational complexity. The quantization step included a posttraining process in which a small calibration subset (see Fig. 2) of training images, i.e., 1000 calibration images, was used to analyze the distribution of activations and to limit the accuracy degradation. Data-free quantization (DFQ) algorithm [35] was used in posttraining process to equalize the weight ranges and to correct biases in the errors introduced during quantization. After

model quantization, each layer is converted by an integrated Vitis AI compiler into a sequence of associated supported instructions that actuate the DPU.

C. Data Processing

For continuous land scanning, the *carpet mapping* imagery requires a large amount of storage space resulting from the combination of swath width, spectral channels count, and spatial resolution of the sensor. For example, Sentinel-2 sensor acquires 1.6 TB of useful data per orbit, resulted from a 290 km swath with 13 spectral channels at a spatial resolution of 10 m. For long-term acquisitions, imagery is too large to be processed efficiently in memory by convolutional operations. Thus, a significant latency between acquisition and result output is generated. To overcome this problem, the band interleaved by lines (BIL) [36] format is used for real-time on-board applications.

The line-by-line technique reduces the on-board memory footprint and output latency making it ideal for satellites equipped with push-broom sensors. In the BIL data format approach, combined with push-broom optical sensors such as those mounted on Sentinel-2 or Landsat 8 satellites, N lines of pixels need to be received in order to prepare a collection of quadratic samples. Additionally, super-resolution methods that enhance spectral measurement by tilting an area sensor instead of shifting a linear sensor can be adjusted to maximize the data throughput for our proposed patch-level method [37]. It is worth noting that although the proposed method is demonstrated just on data acquired by Sentinel-2, the same approach can be extended to other optical remote sensing products that acquire data in the visible, near-infrared and shortwave infrared parts of the spectrum.

IV. STUDY AREAS AND DATASETS

Fire detection systems combine data from RGB cameras, by using color information, and InfraRed (IR) cameras, by measuring the thermal radiation, to achieve a consistent level of accuracy. Preparing multispectral data for training and testing the proposed algorithm requires building and annotating datasets based on off-the-self datasets, such as the Sentinel-2 archive. MultiSpectral Instrument (MSI) mounted on Sentinel-2 satellite measures the earth's reflected radiance in 13 spectral bands, i.e., B01, B02, B03, B04, B05, B06, B07, B08, B8A, B09, B10, B11, B12, from visible (RGB) to short-wave IR (SWIR) [38]. In this study, three datasets from data acquired by the Sentinel-2 mission were studied, named BigEarthNet, Zamora, and Bordeaux. BigEarthNet was operated as a genesis dataset, in different versions, to train a feature extractor model capable of distill multispectral images. Forwards, two proposed datasets, i.e., Zamora and Bordeaux, were employed in the conducted experiments.

Remote sensing images generally incorporate areas with great semantically complex diversity that usually must be expressed through multiple class labels. In order to maintain a high interclass variance, a conglomerate scene produces a substantial collection of samples, i.e., patches. Usually, the samples are

quadratic, with a size directly proportional to the semantic diversification of the original scene. In this work, the patch size considered was 120×120 pixels with ten spectral bands.

A. BigEarthNet Dataset

The BigEarthNet [39], [40] is a large-scale Sentinel-2 benchmark archive, consisting of 590 326 Sentinel-2 image patches. The image patch size on the ground is 1.2×1.2 km² with variable image size depending on the channel resolution. This is a multilabel dataset with 43 imbalanced labels. Each Sentinel-2 image from BigEarthNet dataset was associated with one or more class labels extracted from the CORINE land cover map of 2018, with an accuracy of around 85% [41]. The solar radiation reflected from the earth's surface to the Sentinel-2 sensor is quantified in bands at different wavelengths [42]. As band B10 does not include surface information, it was not included in the BigEarthNet dataset nor in the proposed datasets.

B. Proposed Datasets

The Copernicus Open Access Hub¹ provides complete, free, and open access to Sentinel-1, Sentinel-2, Sentinel-3, and Sentinel-5P user products. From the Copernicus portal, two separate Sentinel-2 products were downloaded, named Zamora and Bordeaux due to the fact that the products were acquired in close proximity to these cities. A Sentinel-2 product includes a multispectral scene with a size of 10980×10980 pixels, with reference to 10 m bands [42]. For each package, a region of interest (ROI) was chosen based on in situ information. The SeNtinel Application Platform (SNAP)² was used to extract the ROI from each product. To successfully crop the ROI, the bicubic method was used on all bands with resolution of 20 and 60 m to up-sample to 10 m. This step was applied to all the data evaluated because, in the context of deep learning, it is advisable to use a homogeneous collection of data to reduce the sensitivity of the model to variations in the distribution of the input data [43].

Next, each scene derived from each ROI was split into nonoverlapping patches, generating a dataset. The image patch size was identical to that in the BigEarthNet dataset, i.e., size on the ground of 1.2×1.2 km², for compatibility reasons in using the model originally trained on the BigEarthNet dataset. Within the proposed datasets, vegetation areas were deemed as the background of the dataset while forest fires affected areas were considered as the anomalies. To generate the ground truth (GT) map, i.e., to identify burned and unburned patches, necessary to evaluate the proposed method, a classical spectral indices method was utilized. The way the classical spectral index method is applied is described in [10]. The proposed datasets were described below. The total number of samples, the number of burned samples, and the ratio of burned samples in the entire dataset were included in Table I. In these datasets, an anomalous sample contains active fire fronts or vegetation already burned across the entire footprint, while a normal sample can consist

¹<https://scihub.copernicus.eu/dhus/#/home>

²<https://step.esa.int/main/toolboxes/snap/>

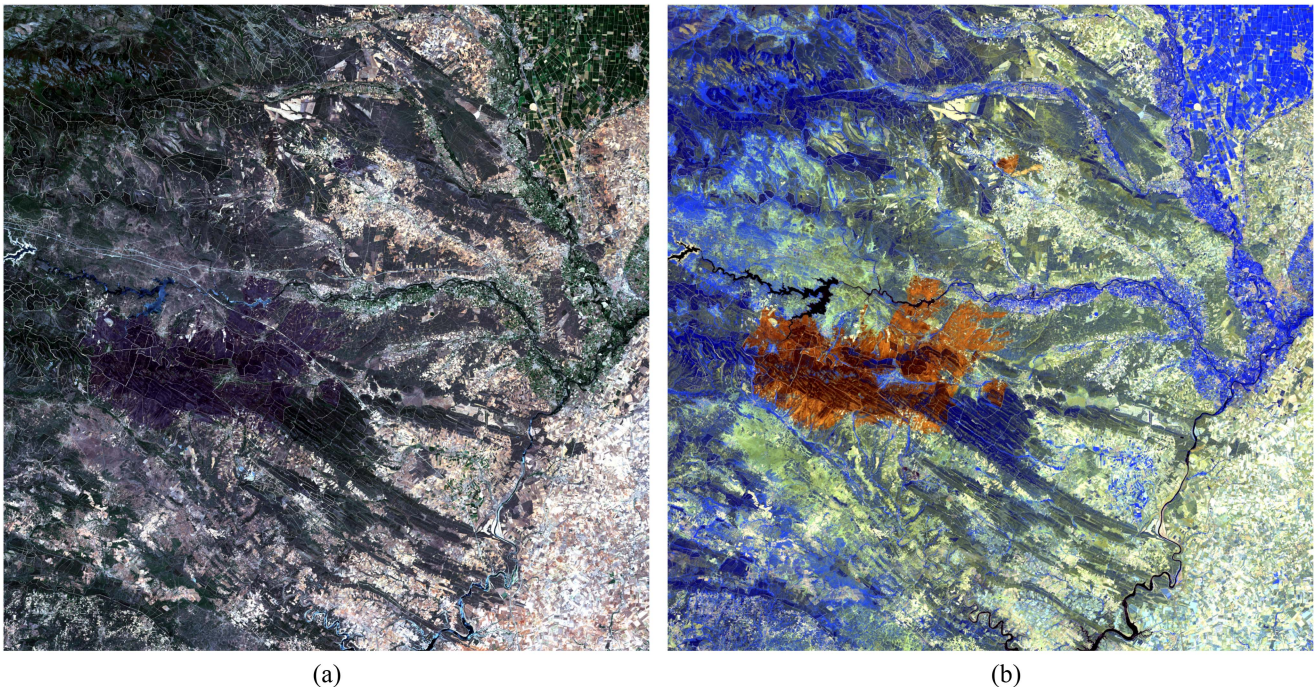


Fig. 3. Zamora scene (Spain, 2022). In (a) is displayed in TCI format (R: Band 4; G: Band 3; B: Band 2) and in (b) is shown in FCI format (R: Band 12; G: Band 11; B: Band 8 A). The burned areas (anomalous samples) are highlighted in brown color in (b).

TABLE I
SUMMARY OF THE USED DATASETS

Dataset	Total samples	Burned samples	Ratio
Zamora	3600	147	4%
Bordeaux	3200	82	2.56%

of any combination of normal classes, i.e., common land cover classes.

1) *Zamora*: The Zamora scene contains disastrous wildfires that took place in Spain during June 2022, in the northwest province of Zamora, mountain range Sierra de la Culebra. This scene was recorded on June 28, when over 30 000 hectares of a highly biodiverse ecosystem were already burnt [44]. During recording, the scene was completely cloud free. A ROI of 7200×7200 pixels (with reference to 10 m bands) was cropped using in situ information. Using an image patch size of 120×120 pixels (with reference to 10 m bands), it results a cardinality of 3600 samples in this dataset. Fig. 3(a) exhibits a true-color image (TCI), i.e., a combination of three channels that are sensitive to the red, green, and blue visible light, for the selected ROI in order to facilitate a quick delineation of surface variety and atmospheric characteristics. In Fig. 3(b) is shown a false-color image (FCI) that renders nonvisible parts of the electromagnetic spectrum, i.e., bands B12, B11, B8A from SWIR range, aiming to visually analyze different enhanced features in the landscape, e.g., burned areas in brown color. The scene contains areas of wheat fields, pastures, vineyards, and wooded hills.

2) *Bordeaux*: The Bordeaux scene contains devastating blazes that took place in southwest France during June, July,

and August 2022, in Gironde department, near Bordeaux city. This scene was recorded on July 17 with clear skies and thin smoke generated by the still-burning canopy. An ROI of 4800×9600 pixels (with reference to 10 m bands) was cropped using in situ information [45]. After performing a nonoverlapping cropping, based on a 120×120 pixel patch, a collection of 3200 samples was obtained. Fig. 4(a) and (b) depicts a TCI, respectively, an FCI for the selected ROI. The composition of SWIR bands in Fig. 4(b) provides a picture of the burned areas and reveals areas of ongoing fire where smoke is not too opaque, as in the highlighted sample. Besides the metropolitan area of Bordeaux, the terrestrial space scene contains hills, pastures, forests, orchards, vineyards, all interspersed with hilltop villages.

V. EXPERIMENTS

Deep learning models are powerful feature extraction methods capable of extracting more abstract and salient features from data, compared to traditional approaches. To obtain a powerful spectral-spatial feature extractor, a ResNet model [46] with 50 convolutional layers was trained on the BigEarthNet dataset. Subsets of the BigEarthNet dataset were considered depending on the specifics of the downstream task. Hereinafter, the ResNet50 model is denoted as a multiclass classifier used to obtain a feature extractor. ResNet-based networks preserve representations from degrading toward the end of the network thanks to the incorporated residual units. The DCNN model was implemented in the Tensorflow framework due to high integration with the software stack responsible for quantizing and compiling the model into DPU instructions. As proposed

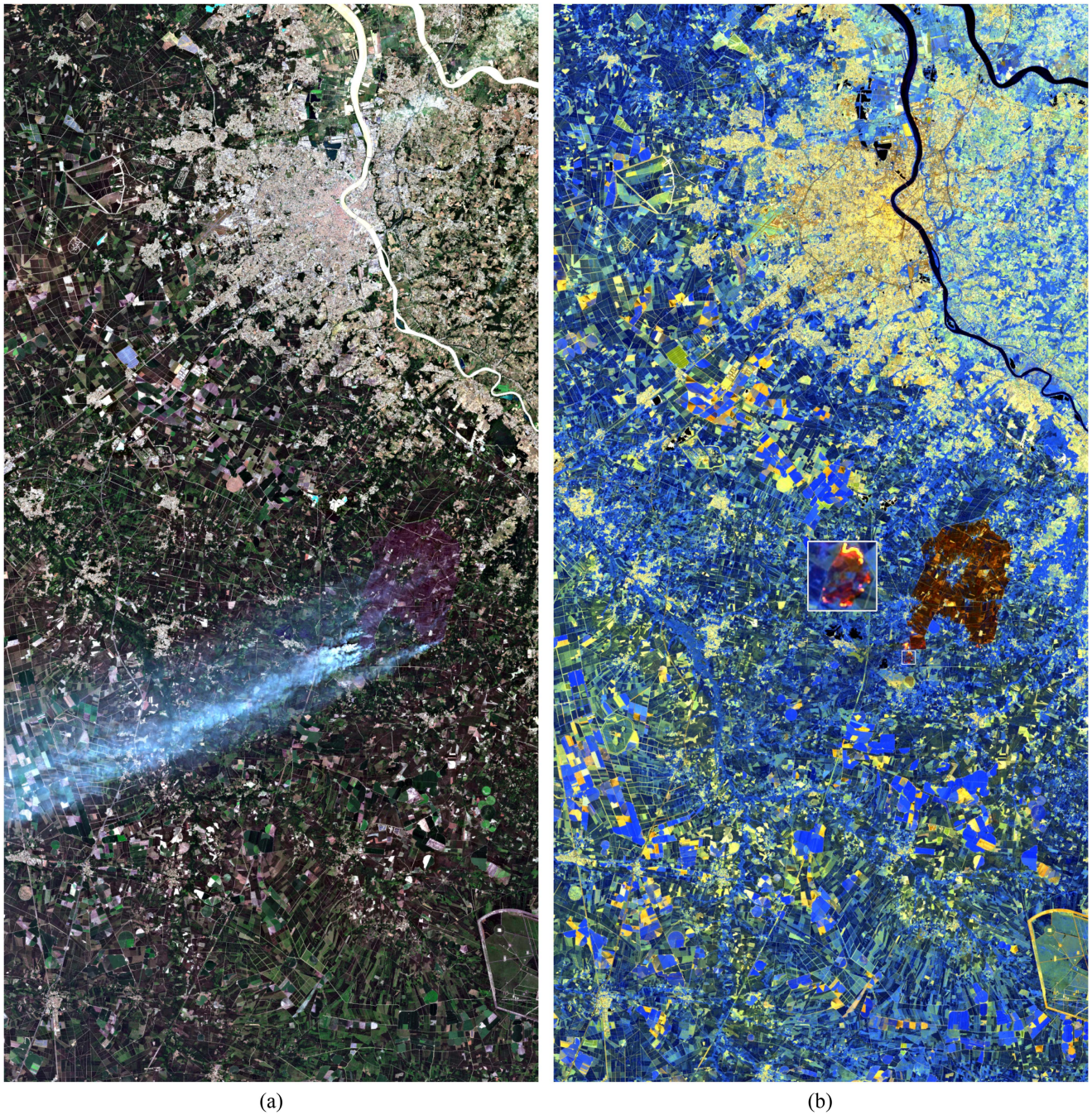


Fig. 4. Bordeaux scene (France, 2022). In (a) is displayed in TCI format (R: Band 4; G: Band 3; B: Band 2) and in (b) is shown in FCI format (R: Band 12; G: Band 11; B: Band 8 A). The burned areas (anomalous samples) are highlighted in brown color in (b). In the highlighted sample, the fire fronts are visible due to the long wavelength bands used in the FCI composite.

in [39], the images from the BigEarthNet dataset that were completely covered by seasonal snow, clouds, and cloud shadows were removed, resulting in a clean version of BigEarthNet. Here and throughout, training datasets were derived starting from the clean version. In the proposed experiments, the studied areas (see Section IV-B) contain one specific anomaly, i.e., burned areas, in Zamora and Bordeaux. To detect anomalous burned areas inside complex land cover multispectral scenes, class *Burnt areas* was filtered out from the clean version of BigEarthNet,

resulting in a 42 class-nomenclature [40]. Afterward, the rule 60:20:20 for randomly choosing training, validation, and test datasets was adopted. During the training, the network received a frame of $120 \times 120 \times 10$ pixels at the input, considering only the bands that originally had resolutions of 10 and 20 m. The GT map used in the training phase was provided by the creators of the BigEarthNet dataset [39]. The adaptive moment estimation method, i.e., Adam, was applied to optimize the trainable network, with a learning rate of $1e - 3$, 100 epochs, and

a batch size of 300 samples. The ResNet50 model was trained on a workstation machine equipped with Intel(R) Xeon(R) CPU E5-2620v4@2.10 GHz and Tesla K80 12 GB GPU. For a dataset of 254 812 samples, the training process took approximately 48 h. The loss function used was a sigmoid cross-entropy loss function, obtaining a minimum value of 0.0132.

After training the multiclass classifier, the MAVs were calculated on the training dataset, for each class, taking into account all samples that were correctly classified (see Fig. 1). Working with a multilabel dataset, a correctly classified sample can be included in the MAV calculation of multiple classes, depending on the number of labels assigned to that sample. After finding the MAVs, the distance vectors were estimated with cosine distance as follows:

$$\cos(\theta) = \frac{A \cdot B}{\|A\|_2 \|B\|_2} \quad (4)$$

where A is the activation vector of an observed sample and B is the corresponding MAV of the class. Further, in the Weibull tail fitting step, the parameter α was fixed to five since 96.80% of images in BigEarthNet dataset are not associated with more than five labels [40].

In on-board implementation, a programmable engine optimized for DNN, i.e., DPU, provided by the Vitis AI development environment for AI inference on Xilinx hardware platforms was used. DPU component uses a specialized tensor-level instruction set that efficiently assists the implementation of DL networks and accelerates the computing workloads of DL inference algorithms. The DL accelerator is placed on FPGA fabric. In the conducted experiments, the target platform is a Xilinx Zynq UltraScale+ MPSoC ZCU102 evaluation board³ which combines a powerful processing system (PS) and a user-programmable logic (PL) into a powerful MPSoC, i.e., Zynq UltraScale+ XCZU9EG-2FFVB1156E MPSoC. This MPSoC runs a custom light-weight Linux-based operating system, i.e., Petalinux,⁴ on the ARM Cortex A53 64-b quad-core multiprocessing CPU. Additional, this MPSoC incorporates also an ARM Cortex-R5 32-b dual-core real-time processing unit. This CPU-FPGA hybrid system was programmed on each part, the CPU with logic for input and output processing, and the FPGA with execution of the DCNN architecture.

The target application is run on the CPU with Xilinx runtime application interface calls to manage runtime interaction with accelerator. During the on-board deployment phase, the pre-computed MAVs and distance vectors are downloaded together with the host application to the target platform and saved on disk (see Fig. 2). MAVs have a static storage size of approximately 17 KBytes, while the distance vectors storage size vary according to the cardinality of the observed dataset, i.e., in these experiments, the size is approximately 24 MBytes. Also, the libMR library [18] was cross-compiled for Arm v8 architecture in order to be used on the host application running on the Cortex-A53

TABLE II
ACCURACY EVOLUTION OF MULTICLASS CLASSIFICATION MODEL ON 1000 TEST SAMPLES

Dataset	Model stage	Precision	Recall	F1-score
BigEarthNet	Frozen model	0.811	0.737	0.772
	Quantized model	0.737	0.674	0.704

TABLE III
DETECTION TIME OF THE PROPOSED ANOMALY DETECTOR FOR A TEST COLLECTION OF 60 SAMPLES

Device	# threads	FPS	Processing time [s]
XCZU9EG MPSoC	1	51.931 ± 0.112	1.155 ± 0.002
	2	84.173 ± 3.141	0.713 ± 0.026
	4	77.813 ± 1.726	0.771 ± 0.017
	6	80.259 ± 1.183	0.747 ± 0.011
Tesla K80 GPU	8	85.677 ± 0.954	0.700 ± 0.007
	multithread	19.861 ± 3.252	3.128 ± 0.744

application processing unit. The loading time of the locally saved Weibull model is fixed and is performed only once in the initialization stage.

At inference time, DPU processes work with the input and output of fixed-point data. To deploy the GPU-trained model on an edge hardware, i.e., FPGA platform, for inference, a reduction in computing complexity was performed. Thus, 32-b full precision (FP32) weights and activation values were quantized to 8-b fixed point (INT8) format. Reduction of bit lengths through quantization, i.e., from 32-b float to 8-b fixed, decreases the memory usage four times. Fixed-point network model required less memory bandwidth, providing higher speed and power efficiency than the floating-point model, but with performance reduction due to quantization noise. Table II shows the decrease in accuracy of the multiclass classifier following quantization. There is a small decrease of 0.068 on the F1-score metric.

Fig. 5 depicts four lines of 60 samples from a line-by-line data processing flow. One line is evaluated on the Zynq UltraScale+ XCZU9EG-2FFVB1156E MPSoC device in 0.7 s. Table III shows the evolution of the processing time when testing the anomaly detection algorithm on a collection of 60 samples. The first column identifies the device on which the algorithm is running. The second column, i.e., the number of threads, identifies the degree of parallelization on the target application run on the processor and the third column, i.e., the FPS, indicates the number of frames per second. Since most spaceborne and airborne spectral systems capture data line-by-line, in the fourth column, i.e., the run time, is shown the time needed to evaluate a line of 60 samples, e.g., one line of patches in the Zamora dataset included in the experiments (see Section IV-B). The values in the third and fourth columns, i.e., mean and standard deviation, are obtained over 20 runs. The best time performance is achieved when the anomaly detection algorithm is run on XCZU9EG MPSoC device with 8 threads on the PS side, achieving a processing speed of 81.018 ns/pixel. Based on the best processing speed obtained in Table III, the Zamora and Bordeaux datasets are processed in 42 and 37.3 s, respectively. The running time on the K80 GPU device is an average of 3.128 s, with standard deviation of 0.744 s over 20 runs. On the Tesla K80 GPU device, the

³<https://docs.xilinx.com/v/u/en-US/ug1182-zcu102-eval-bd>

⁴<https://docs.xilinx.com/r/en-US/ug1144-petalinux-tools-reference-guide/Overview>

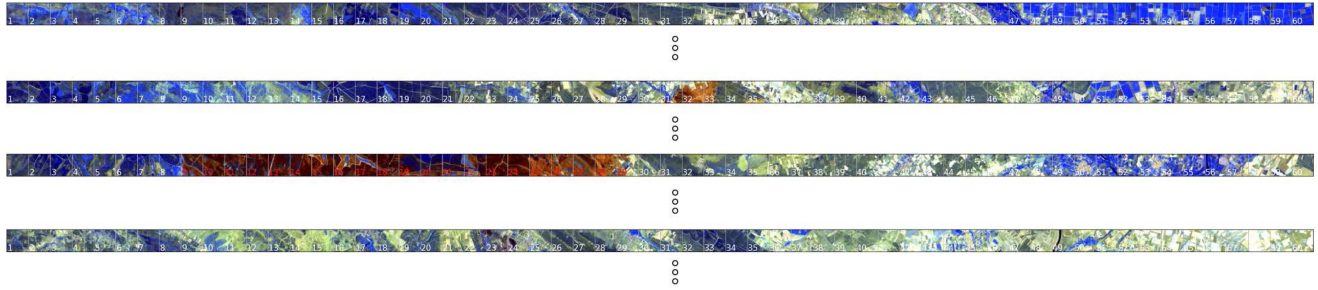


Fig. 5. Processing Zamora scene in band interleaved by lines format (BIL). Each line contains 60 samples shown in FCI format (R: Band 12; G: Band 11; B: Band 8 A) extracted from scene displayed in Fig. 3(b). Each sample has a dimension of 120×120 pixels and the color label indicates if it is anomalous, i.e., red color, or not, i.e., white color.

TABLE IV
COMPUTATIONAL RESOURCES UTILIZATION BY THE PROPOSED ANOMALY DETECTOR ON THE ZCU102 PLATFORM

Resources	LUTs	DSPs	BRAMs	FFs
Used	51351	710	255	98525
Total ⁵	274080	2520	912	548160
Utilization	18.73%	28.17%	27.96%	17.97%

TABLE V
ACCURACY ASSESSMENT OF ANOMALY DETECTOR ON TEST DATASETS

Dataset	Device	Precision	Recall	F1-score
Zamora	Tesla K80 GPU	0.823	0.750	0.784
	XCZU9EG MPSoC	0.727	0.721	0.723
Bordeaux	Tesla K80 GPU	0.829	0.723	0.772
	XCZU9EG MPSoC	0.787	0.683	0.731

running time is about $4.46 \times$ slower than the best time achieved on the FPGA device. Furthermore, the power consumed by the Tesla K80 GPU device alone is 135 W, compared to that of the entire Zynq UltraScale+ board which is 30 W, resulting in an efficiency factor of 4.5 for the latter device.

In Table IV is presented the resource utilization of a DPU (DPUCZDX8G) single core integrated into the proposed anomaly detection accelerator. The B4096 architecture variant was utilized for the DPU configuration to obtain the maximum scaling factor in terms of logic resource utilization and parallelism, i.e., total number of MACs per DPU clock cycle. Although the model used is complex and resource-consuming, it can be limited to using a maximum of one third of the target board resources.

In the context of binary classification, due to the high imbalance in the considered datasets (see Table I), i.e., number of samples in the normal class versus the number of samples in anomalous class differ by more than one order of magnitude, three accuracy metrics were studied, i.e., Precision, Recall, and F1-score. The Precision measure is the rate of predicted anomalous patches that are actual anomalous, while the Recall measure is the ratio of actual anomalous patches correctly identified. The F1-score is a harmonic mean between Precision and Recall. Performance assessment is performed by knowing the GT map of the anomalous samples within the scene under analysis.

Table V depicts the accuracy of the proposed method on datasets described in Section IV-B by running the method on different devices. Due to quantization, there is a small decrease in the F1-score metric when the algorithm is run on the embedded platform, i.e., 0.061 for the Zamora dataset and 0.041 for the Bordeaux dataset. Results in Table V are obtained by setting the anomaly threshold η to 0.5. This means that if, for a query

sample, the highest probability is that of the unknown class, or if the probability of the unknown class is greater than η , then the sample is considered anomalous. Independent of the running device, the proposed algorithm obtained for the Zamora dataset an F1-score value of 0.784 and for the Bordeaux dataset an F1-score value of 0.772. The main parameter that determines the performance obtained by the proposed algorithm is the level of false positive (FP) rate, incorporated in Precision metric. The FP rate, i.e., images that essentially contain no anomalies but are detected as anomalies, is inversely proportional to the Precision metric. The Precision metric obtained by running the algorithm on the FPGA device was 0.727 for the Zamora dataset and 0.787 for the Bordeaux dataset, respectively. In the case of the Zamora dataset, few burnt areas in hilly areas were included in a known class, i.e., *Peatbogs* class. In the case of the Bordeaux dataset, false alarms occurred mostly in drought areas. The outlined sample in Fig. 4(b) is a genuine anomaly because it contains a class that was not seen by the model in the training phase, i.e., fire fronts class. This sample was evaluated by the proposed model, returning an unknown class probability greater than 0.9.

From the results of the proposed experiments, one can see that the proposed algorithm provides a high precision in all cases, keeping false alarms at a low rate. In terms of sensitivity, i.e., recall, the proposed algorithm needs to be improved in background modeling phase. This is a challenging task due to immense spatial coverage with deeply heterogeneous background. There is the well-known problem of label noise from machine learning in remote sensing, which is correlated with the complexity of different applications. Generating perfect labels in remote sensing is impossible, so a trade-off between quantity and quality will always remain in place. One idea for enhancement is to increase the class nomenclature of the training set resulting in

a proportional decrease in intraclass variation, thus increasing the classification accuracy. This improvement, specific to the proposed algorithm, may increase its sensitivity to anomalies in general.

A. Comparison

In order to effectively evaluate the performance of the proposed method, we considered two performance criteria, the hardware resources used, respectively precision and recall metrics. Del Rosso et al. [47] proposed a prototype of an on-board convolutional model for volcanic eruption detection to generate immediate alerts. To demonstrate the method, the authors used an experimental hardware, namely, a drone with a payload composed of a Raspberry Pi 3 Model B processing unit, an Intel Movidius Neural Compute Stick coprocessor unit and a Raspberry Pi camera module (RPi-Movidius system). Multi-spectral data from Sentinel-2 and Landsat-7 satellites were used in the experiments and each scene was divided into 512×512 pixel patches with three spectral bands. Starting from an unbalanced dataset, since the hazard of volcanic eruptions is rare, the augmentation of anomalous samples was used to reach a balanced dataset, following a binary classification. Ultimately, the experiments involved flying a drone over a print made with Sentinel-2 data of an erupting volcano.

The method proposed in this article obtained better results than the method run on the RPi-Movidius system [47] in terms of hardware resources usage and processing speed, while decreasing in F1-score by 0.099. The FPGA method used less than 30% of the hardware resources, while the comparative method made limited use of the resources provided by the RPi-Movidius system. The RPi-Movidius method achieved a processing speed of seven samples (one sample was $512 \times 512 \times 3$ pixels) per second, while our proposed FPGA method attained a speed of 60 samples (one sample was $120 \times 120 \times 10$ pixels) in 0.7 seconds, resulting in an increased speed factor of $2.24 \times$. Considering a tradeoff between performance and speed, the proposed method has a high anomaly detection capability in skewed data collections.

Further, we made a comparison with an anomaly detection system built specifically for an FPGA platform. In [26], a lightweight hyperspectral image anomaly detector for real-time missions was proposed. The authors implemented a pruning–quantization–anomaly–detector (P-Q-AD) for hyperspectral datasets containing anomalies such as ships and planes, and tested it by running it on the same hardware device used in this article, namely, the ZCU102 platform. This allows us to make accurate comparisons in terms of hardware resources used. Since the complexity of the anomalies in the datasets treated by the two papers, i.e., ours and [26], differ considerably, we compared the two methods in terms of detection time per pixel and the amount of hardware resources consumed. We considered the best detection time obtained in [26] for the Sandiego dataset. For a size of $100 \times 100 \times 166$ pixels, the P-Q-AD method required a detection time of 0.56 s, resulting in a processing speed of 337.34 ns/pixel. Our method obtained a processing speed

TABLE VI
COMPUTATIONAL RESOURCES CONSUMPTION BY DIFFERENT ANOMALY DETECTORS ON THE ZCU102 PLATFORM

Detector name	LUTs (274080)	DSPs (2520)	BRAMs (912)	FFs (548160)
P-Q-AD [26]	$2.22e^5$ 80.99%	1092 43.33%	32 3.50%	$2.73e^5$ 49.80%
Ours	51351 18.73%	710 28.17%	255 27.96%	98525 17.97%

of 81.018 ns/pixel on the same hardware platform, resulting in an increased speed factor of $4.16 \times$. A resource consumption is evaluated in Table VI. Due to the reduced data precision, with fewer bits per data, P-Q-AD consumes fewer BRAMs, while more DSPs are employed for 16 b operations and a large quantity of LUTs for network implementation. Resource utilization shows that our implementation confers higher space availability for increased parallelization.

For comparison in terms of precision and recall indicators, we chose a state-of-the-art anomaly detection algorithm [48] that detects postdisaster building damage in different natural disaster contexts, including wildfires. The results of the two methods have a common denominator for comparison because the model used for comparison is strongly influenced by the type and context of the disaster event. In the case of wildfires, the burned building surroundings had a higher discriminative power to derive correct classifications for larger patches due to high anomaly scores, whereas the buildings itself obtained lower scores. Therefore, the Skip-GANomaly location-based model obtained an F1-score of 0.807 on the Santa-Rosa dataset after strict preprocessing, i.e., no vegetation, no shadows, with a patch size of 32×32 . Independent of the implementation device, our method obtained comparable results, achieving an F1-score of 0.784 for the Zamora dataset. However, our results were obtained on a larger patch size, i.e., 120×120 , which implies a larger visual variety and a shorter processing time for the same scene.

VI. CONCLUSION

This work proposed a meta-recognition hardware accelerator for on-board anomaly detection in multispectral imaging missions. The experimental results have shown that the proposed accelerator achieved notable processing speed, i.e., $4.46 \times$ increase, with minimal decrease in accuracy, i.e., 0.041 decrease in F1-score, compared to the equivalent GPU implementation. Integration of AI algorithms next to the sensor significantly reduces the time between image acquisition and image analysis by transmitting only relevant and informative data to the end user, making it possible to produce early warnings and interventions when dangerous events are about to occur.

A future direction for improvement of the proposed solution is to increase the utilization of the FPGA component by running multiple instances of the same kernel, i.e., several instances of the feature extractor that load a set of images simultaneously or at different timings. In this case, a higher throughput can be achieved. At the same time, selecting a reduced number of

multispectral bands used for on-board inference can decrease the computational resources required while maintaining or even improving model accuracy.

The European Space Agency has taken the first steps in deploying AI on-board via the PhiSat-1 nanosatellite mission. Therefore, an on-ground trained model can be deployed and engaged on intelligent systems promoting power-efficient processing platforms. This proposal focused entirely on the implementation of an anomaly detection application running on an embedded device, which could be relevant for future PhiSat missions.

REFERENCES

- [1] H. Hoffmann, "Kernel pca for novelty detection," *Pattern Recognit.*, vol. 40, no. 3, pp. 863–874, 2007.
- [2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.
- [3] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, 2001.
- [4] D. M. Tax and R. P. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, no. 1, pp. 45–66, 2004.
- [5] J. Verbesselt, A. Zeileis, and M. Herold, "Near real-time disturbance detection using satellite image time series," *Remote Sens. Environ.*, vol. 123, pp. 98–108, 2012.
- [6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009.
- [7] S. Matteoli, M. Diani, and G. Corsini, "A tutorial overview of anomaly detection in hyperspectral images," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 5–28, Jul. 2010.
- [8] M. Coca, A. Anghel, and M. Datcu, "Unbiased seamless SAR image change detection based on normalized compression distance," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2088–2096, Jul. 2019.
- [9] A. Rostami, R. Shah-Hosseini, S. Asgari, A. Zarei, M. Aghdami-Nia, and S. Homayouni, "Active fire detection from landsat-8 imagery using deep multiple kernel learning," *Remote Sens.*, vol. 14, no. 4, 2022, Art. no. 992.
- [10] M. Coca, I. C. Neagoie, and M. Datcu, "Hybrid DNN-Dirichlet anomaly detection and ranking: Case of burned areas discovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Sep. 2022, Art. no. 4414116, doi: [10.1109/TGRS.2022.3207311](https://doi.org/10.1109/TGRS.2022.3207311).
- [11] M. Coca and M. Datcu, "Anomaly detection in post fire assessment," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 8620–8623.
- [12] J. Keeley and A. Syphard, "Large california wildfires: 2020 fires in historical context," *Fire Ecol.*, vol. 17, 2021, Art. no. 22.
- [13] 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, Mar. 2018.
- [14] D. Rudolph et al., "CSP: A multifaceted hybrid architecture for space computing," in *Proc. 28th Annu. AIAA/USU Conf. Small Satellites*, 2014, pp. 1–7.
- [15] *RT Kintex UltraScale FPGAs for Ultra High Throughput and High Bandwidth Applications (WP523)*, Xilinx, May 19, 2020, [Online]. Available: <https://docs.xilinx.com/v/u/en-US/wp523-xqrku060>
- [16] G. Giuffrida et al., "The phi-sat-1 mission: The first on-board deep neural network demonstrator for satellite earth observation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.
- [17] G. Giuffrida et al., "Cloudscout: A deep neural network for on-board cloud detection on hyperspectral images," *Remote Sens.*, vol. 12, no. 14, 2020, Art. no. 2205.
- [18] W. J. Scheirer, A. Rocha, R. J. Micheals, and T. E. Boult, "Meta-recognition: The theory and practice of recognition score analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1689–1695, Aug. 2011.
- [19] 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.
- [20] I. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 10, pp. 1760–1770, Oct. 1990.
- [21] Q. Du, B. Tang, W. Xie, and W. Li, "Parallel and distributed computing for anomaly detection from hyperspectral remote sensing imagery," *Proc. IEEE*, vol. 109, no. 8, pp. 1306–1319, Aug. 2021.
- [22] M. Díaz, R. Guerra, P. Horstrand, S. López, and R. Sarmiento, "A line-by-line fast anomaly detector for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8968–8982, Nov. 2019.
- [23] P. Horstrand, M. Díaz, R. Guerra, S. López, and J. F. López, "A novel hyperspectral anomaly detection algorithm for real-time applications with push-broom sensors," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4787–4797, Dec. 2019.
- [24] J. Caba, M. Díaz, J. Barba, R. Guerra, S. Escobar, and S. López, "Low-power hyperspectral anomaly detector implementation in cost-optimized FPGA devices," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 2379–2393, Mar. 2022, doi: [10.1109/JSTARS.2022.3157740](https://doi.org/10.1109/JSTARS.2022.3157740).
- [25] L. Zhang et al., "Fast real-time causal line-wise progressive hyperspectral anomaly detection via Cholesky decomposition," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 10, pp. 4614–4629, Oct. 2017.
- [26] N. Ma, X. Yu, Y. Peng, and S. Wang, "A lightweight hyperspectral image anomaly detector for real-time mission," *Remote Sens.*, vol. 11, no. 13, 2019, Art. no. 1622.
- [27] J. Lei, G. Yang, W. Xie, Y. Li, and X. Jia, "A low-complexity hyperspectral anomaly detection algorithm and its FPGA implementation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 907–921, Oct. 2021, doi: [10.1109/JSTARS.2020.3034060](https://doi.org/10.1109/JSTARS.2020.3034060).
- [28] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, Jul. 2013.
- [29] C. M. Bishop et al., *Neural Networks for Pattern Recognition*. London, U.K.: Oxford Univ. Press, 1995.
- [30] Xilinx, "DPUCZDX8G for zynq UltraScale MPSoCs product guide (PG338)," Jun. 24, 2022. Accessed: Nov. 22, 2022. [Online]. Available: <https://docs.xilinx.com/r/en-US/pg338-dpu/Introduction?tocId=Bd4R4bhnWgMYE6wUISXDLw>
- [31] A. Bendale and T. Boult, "Towards open set deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1563–1572.
- [32] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 427–436.
- [33] W. J. Scheirer, Ed., *Extreme Value Theory-Based Methods for Visual Recognition, Ser. Synthesis Lectures on Computer Vision*. Cham, Switzerland: Springer, 2017.
- [34] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.
- [35] M. Nagel, M. van Baalen, T. Blankevoort, and M. Welling, "Data-free quantization through weight equalization and bias correction," 2019. [Online]. Available: <https://arxiv.org/abs/1906.04721>
- [36] J. R. Jensen et al., *Introductory Digital Image Processing: A Remote Sensing Perspective*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.
- [37] K. Kitano et al., "Super-resolution for a dispersive spectrometer using a tilted area sensor and spectrally varying blur kernel interpolation," *Opt. Exp.*, vol. 29, no. 2, pp. 2809–2818, Jan. 2021.
- [38] E. S.A. (ESA), "MultiSpectral instrument (MSI) overview," 2016. Accessed: Nov. 11, 2022. [Online]. Available: <https://sentinel.copernicus.eu/web/sentinel/technical-guides/sentinel-2-msi-instrument>
- [39] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, "Bigearthnet: A large-scale benchmark archive for remote sensing image understanding," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 5901–5904.
- [40] G. Sumbul et al., "BigEarthNet-MM: A large scale multi-modal multi-label benchmark archive for remote sensing image classification and retrieval," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 3, pp. 174–180, Sep. 2021.
- [41] G. Zeug, S. Tumamos, M. Buchauer, A. Pennec, and H. Dufourmont, "HRL grassland 2018 validation report," 2020. Accessed: Nov. 11, 2022. [Online]. Available: https://land.copernicus.eu/user-corner/technical-library/clms_hrl_gra_validation_report_sc04_v1_5.pdf
- [42] European Space Agency (ESA), "Resolution and swath," 2014. Accessed: Aug. 5, 2022. [Online]. Available: <https://sentinel.esa.int/web/sentinel/missions/sentinel-2/instrument-payload/resolution-and-swath>

- [43] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [44] E. M. F. (EFIMED), "Local initiatives to mitigate wildfire risk in Spain," 2022. Accessed: Nov. 15, 2022. [Online]. Available: <https://medforest.net/2022/06/28/local-initiatives-to-mitigate-wildfire-risk-in-spain/>
- [45] BBC, "France firefighters battle 'monster' wildfire near bordeaux," 2022. Accessed: Nov. 15, 2022. [Online]. Available: <https://www.bbc.com/news/world-europe-62503775>
- [46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [47] M. P. Del Rosso, A. Sebastianelli, D. Spiller, P. P. Mathieu, and S. L. Ullo, "On-board volcanic eruption detection through CNNs and satellite multispectral imagery," *Remote Sens.*, vol. 13, no. 17, 2021, Art. no. 3479.
- [48] S. Tilon, F. Nex, N. Kerle, and G. Vosselman, "Post-disaster building damage detection from earth observation imagery using unsupervised and transferable anomaly detecting generative adversarial networks," *Remote Sens.*, vol. 12, no. 24, 2020, Art. no. 4193.



Mihai Coca received the B.S. and M.S. degrees in computer science and information security from the Military Technical Academy "Ferdinand I" Bucharest, Romania, in 2015 and 2017, respectively, and the Ph.D. degree in electronic engineering, telecommunication, and information technologies from the University Politehnica Bucharest (UPB), Bucharest, Romania, in 2023.

He joined CEOSpaceTech as a Research Assistant, in 2017. Since 2015, he has been an Associate Teacher with the Military Technical Academy "Ferdinand I"

covering subjects such as architecture of computer systems and microprocessor systems. In 2018, he performed a three-month internship with the German Aerospace Center (DLR), Oberpfaffenhofen, Germany, as a visiting Ph.D. student with the Remote Sensing Technology Institute (IMF). In the same year, he participated on BigSkyEarth Training School, which took place at the Vicomtech Research Center, San Sebastian, Spain. His current research interests include on-ground and on-board artificial intelligence methods for detecting changes and anomalies in multispectral and SAR imagery.



Mihai Datcu (Fellow, IEEE) received the M.S. and Ph.D. degrees in electronics and telecommunications from the University Politehnica Bucharest (UPB), Bucharest, Romania, in 1978 and 1986, respectively, and the habilitation *a Dirigent Des Recherches* degree in computer science from the University Louis Pasteur, Strasbourg, France, in 1999.

From 1992 to 2002, he had a longer Invited Professor Assignment with the Swiss Federal Institute of Technology (ETH Zurich), Zurich, Switzerland. He was a Visiting Professor with the University of

Oviedo, Oviedo, Spain; University Louis Pasteur, Strasbourg, France; International Space University, Strasbourg, France; the University of Siegen, Siegen, Germany; University of Innsbruck, Innsbruck, Austria; University of Alcalá, Alcalá de Henares, Spain; University Tor Vergata, Rome, Italy; the Universidad Pontificia de Salamanca, Madrid, Spain; University of Camerino, Camerino, Italy; and the Swiss Center for Scientific Computing, Manno, Switzerland. Since 1981, he has been a Professor with the Department of Applied Electronics and Information Engineering, Faculty of Electronics, Telecommunications and Information Technology, UPB. Since 1993, he has been a Scientist with the German Aerospace Center (DLR), Wessling, Germany. He is developing algorithms for model-based information retrieval from high-complexity signals and methods for scene understanding from very-high-resolution synthetic aperture radar (SAR) and interferometric SAR data. Since 2011, he has been leading the Immersive Visual Information Mining Research Laboratory, Munich Aerospace Faculty, and he is the Director of the Research Center for Spatial Information, UPB. Since 2001, he had been initiating and leading the Competence Center on Information Extraction and Image Understanding for Earth Observation, ParisTech, Paris Institute of Technology, Telecom Paris, Paris, France, a collaboration of DLR with the French Space Agency (CNES). He has been a Professor with the DLR-CNES Chair, ParisTech, Paris Institute of Technology, Telecom Paris. He has initiated the European frame of projects for image information mining (IIM) and is involved in research programs for information extraction, data mining and knowledge discovery, and data understanding with the European Space Agency (ESA), NASA, and in a variety of national and European projects. He and his team have developed and are developing the operational IIM processor in the Payload Ground Segment systems for the German missions, TerraSAR-X, TanDEM-X, and the ESA Sentinel-1 and Sentinel-2. He is a Senior Scientist and the Data Intelligence and Knowledge Discovery Research Group Leader with the Remote Sensing Technology Institute, DLR and involved in the DLRONERA Joint Virtual Center for AI in Aerospace. His research interests include explainable and physics aware artificial intelligence, smart sensors design, and quantum machine learning with applications in earth observation.

Prof. Datcu is a Member of the ESA Working Group Big Data from Space and Visiting Professor with the ESA's Φ -Lab. He was the recipient of the Best Paper Award and the IEEE Geoscience and Remote Sensing Society Prize, in 2006, the National Order of Merit with the rank of Knight, for outstanding international research results, awarded by the President of Romania, in 2008, and the Romanian Academy Prize Traian Vuia for the development of the SAADI image analysis system and his activity in image processing, in 1987. He was also the recipient of the Chaire d'excellence internationale Blaise Pascal 2017 for international recognition in the field of data science in earth observation. He has served as a co-organizer for international conferences and workshops and as Guest Editor for a special issue on AI and Big Data of the IEEE and other journals. He is a representative of Romanian in the Earth Observation Program Board.