

PIL TESTING OF THE OPTICAL ON-BOARD IMAGE PROCESSING SOLUTION FOR EO-ALERT

Francisco Membibre<sup>1</sup>, Antonio Latorre<sup>1</sup>, Alexis Ramos<sup>1</sup>, Juan Ignacio Bravo<sup>1</sup>, Robert Hinz<sup>1</sup>, Álvaro Morón<sup>1</sup>, Murray Kerr<sup>1</sup>

<sup>1</sup> DEIMOS Space S.L.U., 28760, Tres Cantos – Madrid, Spain,  
Email: {francisco.membibre, antonio.latorre, alexis.ramos, juan-ignacio.bravo, robert.hinz, alvaro.moron, murray.kerr} @ deimos-space.com

**ABSTRACT**

This paper presents the implementation and verification of the EO-ALERT optical product processing chain, which is a European Commission H2020 project coordinated by DEIMOS Space, where the main objective is to provide a very low latency (<5 minutes) Earth Observation service globally. This is achieved by processing satellite sensor data on-board the flight segment, to obtain derived products, such as alerts in ship detection and extreme weather applications, which can be transmitted via global communications links. The on-board processing is achieved through the efficient use of novel and advanced technologies, including spin-in from other sectors, such as advanced SRAM Multi-Processor, FPGA multi-board reconfiguration and COTS. The HW architecture allows the reconfigurable use of 1 to 4 boards, in a master slave configuration, for this on-board processing, thus reducing the total latency of the product generation for variable-size target areas. On each board the processing takes advantage of the board's resources to obtain the processed products and alerts, which are centralised in the master board. To support the implementation and verification, a testbench is created. The aim of the test bench is to have a multi-board breadboard representative of the final architecture, where the development performance can be verified. The Artificial Intelligence (AI) and Machine Learning (ML) algorithms developed in the project are implemented by migrating them to the target Hardware, taking into account the most efficient implementation on the system multi-core and the parallelization of processing between the available processing boards. The evaluation of the resulting processing system is performed experimentally using real Earth Observation data from a reference-image database, corresponding to the DEIMOS-2 VHR optical satellite and the SEVIRI instrument on the MSG satellite, therefore the results are highly significant. The execution times are measured in the testbench platform, obtaining at the present moment results within the requirements established in the project.

**1. INTRODUCTION**

This paper presents the status of activities performed in the framework of the EO-ALERT Horizon 2020 project, which is composed by 6 partners: DEIMOS Space (Leader), DLR, OHB Italy, Politecnico di Torino (POLITO), TU-GRAZ

and DEIMOS Imaging. Its main objective is to produce on the FS advanced Earth Observation products (e.g. alerts, such as ship detection) obtained from the processing of sensor information (Optical or SAR in EO-ALERT), and transmitting them to the GS and/or to the End User, with very low latency.

Nowadays, there are several services in Europe and USA which are capable of obtaining civil alerts based on EO information, with latencies of 15 min to 30 min. The idea of EO-ALERT is to reduce this latency, processing on-board the images to generate the products on the FS. In the project this is demonstrated on two main application cases, as described below, but the architecture and concept are applicable generally to other scenarios and types of products. This paper presents the two real scenarios developed for Optical multi-band images: ship detection and extreme weather nowcasting. The ship detection scenario works is motivated by the EMSA vessel detection service. It provides for the detection all the ships present in scene (and raw data) acquired by the optical sensor, producing and transmitting ship alerts with a set of associated details, such as its geolocation, size, etc. In EO-ALERT, these alerts are verified via comparison with the known vessel information on ground. This kind of application can be used for illegal fishing, illegal immigration, search and rescue, etc.

In the ship detection scenario, the minimum area coverage that can be considered as a viable product is identified as the min strip scenario, and it corresponds to at least 100 km<sup>2</sup> of maritime area, at 1m/px resolution. Larger areas are also considered, including the so-called max strip scenario, corresponding to the processing of a strip of at least 1000 km<sup>2</sup> at 1 m/px resolution. Our requirements in terms of latency are to cover the min strip scenario in less than 5 minutes, considering the complete process, since the image acquisition until its reception by the user. This effectively implies to be able to process the Earth Observation sensor information in less than 2 to 3 minutes, considering to time for transmission to the end user globally.

The extreme weather scenario is oriented to forecast quick evolution storms that can generate natural disasters and/or cause problems to the general population. A rapid alert is essential to decrease the potential damage caused by the storms, or to activate support and rapid response teams. In this scenario, it is covered 1.200.000 km<sup>2</sup> in less than 5 minutes as well, considering the sensor in the SEVIRI

instrument on the geostationary MSG satellite. To accomplish with the proposed latencies, some parts of the conventional processing chain that normally are located on ground (classical image processing, ML and AI) has been moved to on-board. Figure 1 depicts the changes related to the new processing chain in red, showing that the raw data generated from the sensors is processed on board and the system is able to detect the products of a specific application. To find these possible cases, the on-board processing chain includes AI, using different techniques to obtain the best possible results. The use of these techniques on-board is becoming more common, enabled by the use of powerful COTS processing units in space systems, overcoming its limitations through the application of multiple mitigation techniques.

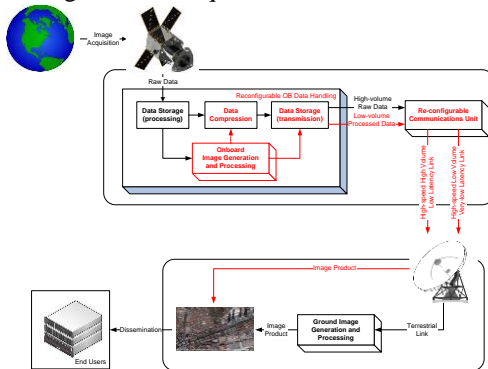


Figure 1. Next generation satellite processing chain for rapid civil alerts.

Finally, EO-Alert presents a quad-board scheme: one board will be the Compression, Encryption and Data Handling (CEDH) board, and the remaining boards will be for processing tasks: one Master Processing Board (MPB), managing the processing operations and two Slave Processing Boards (SPB). The number of processing boards used for Optical processing will be configured depending on the size of the target area or the latency time requirements. The processing of the max strip scenario, with over 1000 km<sup>2</sup>, will take advantage of all the boards.

## 2. ON BOARD ARCHITECTURE

The base processing unit used for the EO-ALERT architecture is the Zynq® UltraScale+™ MPSoC ZU19EG from Xilinx® which includes 4 Cortex™-A53. This kind of boards is broadly used in many areas due to the flexibility and rapid development allowed by the processors, as well as the computing power and parallelism provided by the FPGA hardware. Despite of being COTS hardware, the Zynq® UltraScale+™ and especially the Zynq®-7000, is used in many Low Earth Orbit (LEO) satellites, introducing elements against radiation [2] as described in section 6. Generally, these devices are composed by two main parts:

the programmable logic (PL), where the FPGA and all its elements (LUTs, registers, BRAMs, DSPs, etc.) are located., and the processing system (PS), composed of processors, GPU, memory and interfaces. To communicate the PL with the PS and, the AXI protocol will be used.

### 2.1 Programmable Logic – Platform HW Component

The EO-ALERT architecture uses the ZU19EG FPGA, the largest of the Zynq® UltraScale+™ family. Furthermore, the SDSoc™ provide high-performance resources and communication buses.

Taking into account these capabilities, some parts of the project design are implemented in the FPGA, including utility and processing cores. PCIe protocol implementation is one of them, through its driver and specifying the peripheral, it allows to make high-speed data transfers between boards. This interface works as a master-slave model, where the Root Port (RP) will be the master and the slave will be the End Point (EP). The master RP will write/read to/from the slave EP PL-DDR memory, and the EP will just manage the reading of data and the allocation of results from/to EP PL-DDR. Regarding the multi-board scheme used in EO-ALERT, CEDH will be RP to the MPB's EP. In addition, MPB will be RP for the communications with the SPB (the MPB has 2 PCIe interfaces to make this possible). Finally, the SPB will always be EP for the MPB-SPB communications. More details about PCIe interface will be explained in section 3.1.1 below.

To move data between PS and PL memories through AXI interfaces without interrupting the processors, a DMA is used. Regarding the PL-DDR memory, a 5 GB DDR4 SDRAM with Error Detection And Correction (EDAC) code is implemented. The data sent or received through the PCIe interface is stored on this memory, which is connected as a peripheral, and therefore will be managed through a device-driver. The data received through the PCIe bus transfers can also be stored at the PS-DDR memory (memory assigned to the PS).

### 2.2 Processing System – Platform SW Component

From the PS processing system, the 4 available ARM A53 processors, with 4 GB DDR4 ECC SDRAM, will be used for applications. The large amount of RAM is a big advantage for the described applications, due to the big size of the raw data processed (consider for instance the size of a raw data image of 100 km<sup>2</sup> at 1m/px). The Hardware and system configuration are described in a Hardware Description File, which is used with the PetaLinux tools for generating a custom Linux taking into account the elements of the PL, memory, storage, etc. With PetaLinux it is possible to customize and create an embedded Linux for the PS, modify the boot loader or the kernel, and include

applications. Moreover, designers can merge and adapt the software platform with the hardware previously defined, increasing the performance of the system. Having an OS like Linux on-board increases the flexibility of the system, handling and managing the system from a high level point of view and providing a relevant amount of pre-existing software libraries for multiple purposes.

### 2.3 Platform

The low-level design of PL and PS is elementary to develop the application on the top. Once both parts are designed, it is possible to create the platform. In this project, a custom board design is needed for the custom board. The platform is used to merge the PS and PL resources and provide a complete board base design. To implement the algorithms on top of this base design, the Xilinx® SDSoc™ tool is used. This tool simplifies the process to create ML and AI applications for on the edge applications, taking advantage of parallelism and computing power of hardware. However, although these hardware libraries are programmed with a high-level language as C++, hardware knowledge is still required to avoid typical limitations and constraints of the Hardware based developments, such as fixed-point representation problems, which will be explained in section 5 below.

## 3. MULTI-BOARD SCHEME AND COMMUNICATIONS

To increase the system performance, a multi-board scheme is proposed, so that larger amounts of raw data can be processed in parallel, by sharing data-chunks between the available boards. To transfer this large quantity of data, the PCIe protocol is used. The MPB will manage the raw data received from the CEDH board, sending fragments of this data to the several SPB available, for distributed processing. To control the high number of transfers, the master-slave model and the system status, a telecommand protocol is introduced between the system boards. Section 3.1 below describes the inter-board interfaces and the data/information shared.

### 3.1 Interfaces

As said before, all boards are connected through several interfaces. The following subsections are focused on PCIe and Ethernet interfaces, and the data transferred through them.

#### 3.1.1 PCIe Interface

This interface is used to transfer between boards all data required to run the algorithms. Given the large amount of data shared during the processing of a specific acquisition a fast interface is important. Because of this reason, the PCIe is a great option.

Following the acquisition from the sensor, the CEDH obtains the raw images, sharing them with the MPB. Depending on the area covered, the MPB decides how many data-chunks it accepts and how many will be distributed between the available SPB boards. When any SPB finishes the processing and obtains results, it requests to the MPB to send the concrete products. Later on, the MPB will join and consolidate the products associated to the full raw data and will request to the CEDH board to send the final processing results.

#### 3.1.2 Telecommand link

To handle all these PCIe transfers between boards, different telecommands are defined. To send and receive them, the TCP/IP protocol is used. This bus will be replaced in the final system by a CAN bus, due to its reliability and space-readiness. The reason of introducing these telecommands is to avoid overwriting and collisions when PCIe transfers are done, preventing problems when multiple slave boards intend to write or read products to the MPB or the CEDH board, or in situations when write and read operations coincide at the same time and could overwrite the information stored in the PCIe allocated memory. For instance, an RP could write to the EP PL-DDR whereas the EP is reading a previous data transfer sent to it. In these situations, the exchange of telecommands advises the transmitting or receiving ports and boards of the intended transfer and allow for a traffic control mechanism, giving these boards the possibility to lock unintended transfers and overwriting of the information, until the previous information has been handled correctly and the bus memory areas have been freed. Basically, two types of handshake shown in Figure 2 are defined in function of the type of operation.



Figure 2. RP to EP (left) EP to RP (right).

To make this communication possible and use the Ethernet interface, it is necessary to establish a client-server model between boards with sockets. To manage it, one socket is created between each part of the system boards. In the boards, 2 threads per socket are created, where one of them performs the read operations and the remaining thread does the write operations. The communication between threads is managed by blocking queues. Section 5 explains the different processes running in parallel and how this system is handled. Since the MPB is connected with 3 boards (CEDH and 2 SPB), 3 sockets are created to communicate with each one.

### 3.2 Multi-board scheme

Multiple boards will be used to process in parallel more quantity of data or to reduce the latency. A minimum of two boards (and a maximum of four) is considered in the present design. The minimum of two boards covers the CEDH board (for handling of the instrument data and of the processing subsystem) and an Optical processing board. More boards can be added to distribute and extend the optical data processing. The boards are interconnected through the Ethernet bus (in the future CAN bus) and PCIe interfaces. It is interesting to note that the PCIe interfaces are point to point links, connecting the master CEDH with the master processing board, and this one with the slaves.

## 4. PROCESSING ALGORITHMS

This section will describe the algorithms of the two optical processing applications developed in the project.

### 4.1 Ship Detection

The Optical ship detection scenario comprises two main objectives. The generation of an on-board L1B image product, and the processing of the on-board generated image with ML / AI techniques to obtain a ship detection product.

The on-board L1B image consists of:

- 1) High resolution calibrated and denoised image
- 2) Geolocation information
- 3) High resolution sea-land binary mask (3m/px)

To generate the L1B product, first the raw data obtained with the payload is calibrated to remove pixel inconsistencies and to convert the pixel digital counts to radiances. The calibrated image is then processed with an edge-aware denoising algorithm based on optimized convolutional operations.

The position provided by the satellite GPS and the attitude provided by the AOCS are used to geolocate the image corners. The geolocation algorithm is prepared to integrate Earth Orientation Parameters (EOP) that can be sent to the spacecraft. The geolocation information is used to retrieve the sea-land pixel binary mask that is efficiently stored on-board using a compression algorithm. To avoid decompressing the whole land-mask on-board, which could lead to memory issues, the minimum chunk of information required to generate the sea-land mask for the image is extracted and decompressed on the on-board memory. Using projective transformations, the sea-land mask is projected onto the image coordinates to assign sea-land information to each of the image pixels.

The process of detecting ships over the generated L1B image is based on a three-step approach:

- 1) Candidate ship extraction
- 2) Ship discrimination

### 3) Fusion

The first step, candidate ship extraction, is designed to obtain salient regions of the image that have ship alike shapes and intensities. Otsu thresholding [4] combined with intensity and shape metrics are used to create a binary image containing the areas likely to contain a ship. This information is fused with the land-sea mask to remove salient areas detected over land regions. This step, opposed to sliding-window techniques that classify each region of the image, reduces the search space leading to a more efficient solution for on-board execution at the cost of not providing constant processing times.

In the ship discrimination step, each image region from the binary image is labelled independently using the algorithm from [5]. The regions from the image, corresponding to the labels from the binary image, are classified using a ML approach, combination of Gabor features and Support Vector Machine to assess the presence of a ship.

The last step, fusion, is designed to remove ships detected on overlapping areas of the image (in case the image is divided over different boards) or to suppress different detections of the same object (non-maximum suppression). The process of image generation, saliency mask creation and detection are presented in Figure 3.

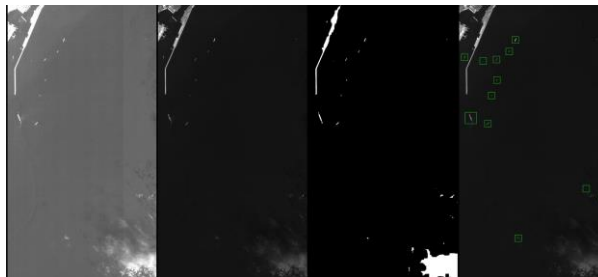


Figure 3. Process of image generation, saliency mask creation and detection. From left to right: raw data, generated image, saliency mask, with landmask, result of the detection.

### 4.2 Extreme weather

The EO-ALERT Optical extreme weather product exploits on-board low latency processing for very short-range forecasting of convective storms. Using a 3-step algorithmic approach, convective storms are identified in GEO satellite Multi-spectral images (Figure 4): Candidate convective cells are detected as local temperature minima in brightness temperature images derived from satellite data using optimized computer vision techniques. Cells are then tracked over time to monitor their displacement and evolution by matching candidate cells between images of subsequent acquisition times based on spatial overlap in the corresponding candidate maps. Finally, ML classifiers are used to determine the convective state of each candidate

cell. Cell features for automatic classification are derived from brightness temperatures in 5 infrared channels in Meteosat's SEVIRI imagery and their respective evolution for each cell over time by calculating a series of statistics (e.g., minimum, mean, maximum), the respective intra-channel differences between acquisition times and inter-channel differences for same acquisition times. Ground truth data for training of the machine learning algorithm is generated from composites of Meteosat-SEVIRI images and OPERA weather radar network data.

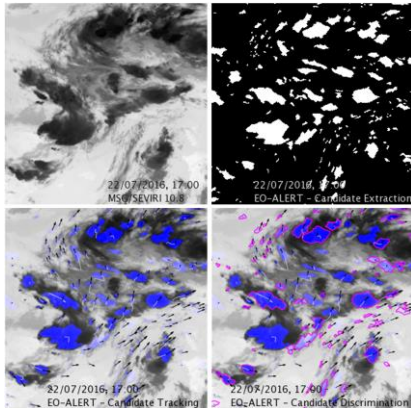


Figure 4. Extreme weather processing steps.

## 5. ALGORITHMS IMPLEMENTATION (HW-SW CODESIGN)

As stated before, to implement on board these algorithms, SDSoC™ and the platform created are used. These tools support the codesign of the implementation of the processing algorithms, distributing them between software and hardware components. The codesign method greatly reduces developed time. It is required to take into account the type of algorithm being applied, and its main characteristics, to identify the characteristics that are better suited for Hardware or Software and to allow obtaining the highest performance. To choose the best implementation, the different algorithms were measured in terms of time and accuracy. Functions whose complexity imply a large amount of processing time are candidates to migrate them to the FPGA. However, for each case it is required validate if the algorithm is suitable for FPGA implementation with a reasonable effort and efficiency gain. As mentioned in subsection 2.3, with the platform it is possible to program the FPGA through high level languages. Some hardware functions are included in SDSoC™, concretely and focusing on image processing, Xilinx® implements some OpenCV functions besides other kind of libraries.

In order to avoid dependencies between FPGA and processors, it is important to allocate hardware functions together, reducing or suppressing the movement of data back and forth from FPGA to processors and vice versa.

Before implementing these functions, a fixed-point model is done with tools such as MATLAB®. This type of model avoids overflow and unalignment problems. Once the model is validated and the accuracy obtained is within range, the inputs are migrated in fixed point representation to the SDSoC™ tool. When FPGA implementation is not feasible, the multi-core option must be evaluated. These resources are very useful when an algorithm is applied over the entire image or in loop. To use the several cores that compose the PS, OpenMP is chosen to increase the performance. The main idea with these resources is to divide the image data into several fragments and apply on them the processing algorithms in parallel, assigning each one to idle cores. This multicore processing is based on fork-join technique, separating application flow to join it when possible and continue the normal execution flow until another fork. Once the code is generated and the required modifications are done, unit tests were done comparing both results, the Hardware implementation (FPGA or CPU) and the preliminary model implementation tested in the design platform by the design expert.

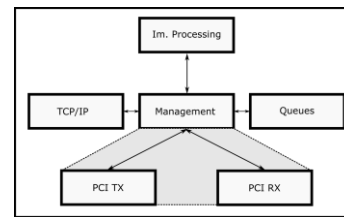


Figure 5. FSM tasks.

## 6. APPLICATION MANAGEMENT

All the execution workflow is based on a set of status which represent a particular stage of an acquisition scenario. In addition to it, a set of telecommands has been designed with the goal of establishing a proper communication system between the different boards. The software system is divided into five different tasks, in which each of them performs a particular operation from the execution flow. The execution workflow of these tasks and their communication is controlled by a Finite State Machine (FSM). The main functions are the following: initialization, FSM control, PCI transmission and reception, Telecommand transmission and reception (with socket handling and telecommand queue) and image processing. In order to improve the performance of the system, the software system implements each of those tasks in a separate thread. The main task defines the basic configuration parameters and spawns the threads of the rest of the tasks. The FSM thread controls the execution flow and the synchronization/communication between them. Each task is connected to the FSM control through a system of blocking queues. Because each state of the FSM defines



a particular stage of a scenario, any operation that does not belong to the current FSM state is neglected. Figure 5 depicts the different tasks.

## 7. SPACE PROTECTIONS

Electronics systems, in particular those used in space missions, are susceptible of suffering the effects of the radiation. These effects can cause a Single Event Effect (SEE) [6]. The most typical events caused by the effects of the radiation are the Single Event Latch-up (SEL) and the Single Event Upsets (SEUs). The current prototype is not protected against SELs, since the protection cannot be applied at design level. Nevertheless, to protect the presented system, a microcontroller is used in background to control the telemetry of the system. If it detects a Latch-Up, then it will trigger the power circuit that will switch off the affected board. The circuit itself is classified as space grade. The Single Event Upsets (SEUs), which are a particular kind of error that modifies the content of the information stored in the system. Since, the proposed system is implemented using a SoC architecture, it must be protected against all the failure types (ASIC and FPGA) since both technologies suffer from different types of errors. Several protection techniques have been applied, such as modular redundancy, Error Correction Codes (ECC) and memory scrubbing.

The redundancy is applied at gross-grain level. The final system will have an additional and configurable spare board like the ones that compose the testbench. In case that one of those boards suffers from a critical or non-recoverable error, this additional board can be configured to replace the faulty one. Moreover, the system is reconfigurable and new bitstream could be loaded from ground, allowing to adapt the system to altered conditions and correcting potential radiation induced errors.

The memories (the DDR4 and the hard disk) are protected against bit-flips by ECCs [10]. In addition, the non-volatile memory (an SSD in the testbench) implements redundancy at logical level, making copies of the content files to assure that there is always an available backup. Furthermore, the telecommands sent between the boards are sent under reliable protocols, covered with CRC and ensuring the data order and delivery. Moreover, the final version will use CAN bus protocol and physical link, which is more suitable for the environment conditions of the system.

The FPGA configuration memory is the largest one on an FPGA and is the most susceptible to suffer the effects of an SEU on it. To protect it, the memory scrubbing technique has been used. This procedure checks periodically the integrity of the configuration memory (typically analysing the CRC). It has been implemented using the Soft Error Mitigation IP core (hereafter called SEM IP) created by Xilinx® for their products [13]. This IP is relatively

customizable and can be configured to detect, detect and correct and inject errors in the configuration memory, as well as classify they type of error.

In the presented system, SEM IP has been configured to classify detect and correct the errors (when possible). If an error is detected but cannot be corrected, the FPGA can be reprogrammed. In the worst case and in the final scenario, the logic of this board could be transferred to the spare board cited before. The rest of the memories from the FPGA (user memories) are protected by ECC, either at implementation or application level.

The reason to use SEM IP to protect the design is that it takes advance of the built-in silicon primitives from Xilinx®, improving this way the performance and efficiency of the protection.

## 8. TESTING

To test this application, real Earth Observation data from a reference-image database are used, corresponding to the DEIMOS-2 VHR optical satellite and the SEVIRI instrument on the MSG satellite, so that the obtained results are highly significative in terms of time and precision. Ground truth information from multiple sources is used in the verification phase.

### 8.1 Optical Ship detection - Min Strip Scenario

This scenario covers ship detection for a scenario of at least 100 km<sup>2</sup> at 1 meter per pixel resolution. A dual-board scheme is proposed, 2 boards (the CEDH board and the MPB) are enough to accomplish the requirements established in the project. In this situation, just the MPB will process the images to obtain the target products, being the CEDH board the master RP and the MPB the slave EP. Additional processing boards can be used to reduce latency, but it is considered relevant to evaluate the latency for this minimum hardware configuration in the min strip scenario. In the Min Strip scenario, the CEDH board takes the images obtained by the DEIMOS-2 VHR satellite and sends them to the MPB. In addition, it sends an ancillary data set containing the GPS coordinates and attitude information obtained during the image acquisition, taken by the satellite. When the MPB board has received these files, it triggers the image processing algorithm. The purpose of the algorithm is to find the candidates (ships) in the provided image. If it detects ships in the image, the system geolocates each of them by using the GPS coordinates provided in the ancillary data. Finally, the alerts and L1B (generated image, optional) are produced and are sent to the CEDH board, from where they will be compressed, encrypted and sent to ground through a low-latency / global coverage network from the satellite to ground.

### 8.2 Optical Ship detection - Max Strip Scenario

This scenario covers ship detection for a scenario of at least 1000 km<sup>2</sup> at 1 meter per pixel resolution. To accomplish the latency and timing requirements established in the project, a quad-board scheme is proposed. This approach maintains the boards presented in the Min Strip scenario, but adds two slave processing boards attached to the MPB board. These two additional boards allow to parallelize the image processing, by doing it through three different boards instead of one. The boards are connected between them in order to establish the telecommand link. The PCIe link use is extended here to use the four boards of the system. The CEDH and master boards are connected the same way that was presented in the Min Strip scenario. However, since there are now four boards, the PCIe ports of the MPB are connected to the slave boards. Figure 6 presents this scheme. In this scenario, several raw image fragments are sent from the CEDH board to the MPB. This board will forward some of the raw data fragments to the SPBs. Data traffic control is implemented by the MPB, interrupting the distribution of data fragments to the slave boards while they are processing.

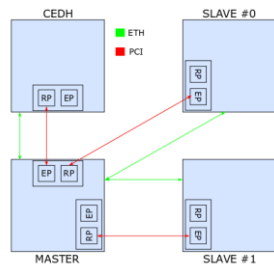


Figure 6. Quad-board scheme.

When the image processing finishes, the master board receives and merges the generated alerts and send them in a unique package to the CEDH board. It does the same with the generated products. When all these packages are received by the CEDH board, the master board accepts the rest of the incoming raw data chunks to process them, starting again all the cited workflow.

## 9. RESULTS

In this section will be presented the results of the project in terms of precision results, showing the detection percentage and the time elapsed to process the images.

### 9.1 Precision Results

For the ship detection scenario, the **preliminary results (prior to tuning)** are obtained over a dataset composed of more than 40 scenes captured with the Deimos-2 VHR satellite. The dataset contains 690 ships of varied sizes on different sea conditions, strong presence of clouds and port areas. The results in terms of Probability of Detection

(POD), False Alarm Ratio (FAR) and F1-score, F2-score are presented in **Table 1**.

Table 1. Preliminary Ship discrimination results.

POD	Recall	FAR	F1	F2
0.757	0.767	0.243	0.762	0.765

For the extreme weather scenario, **preliminary discrimination results (prior to tuning)** in terms of Probability of Detection (POD), False Alarm Ratio (FAR) and F1-score are shown in Table 2. Separate classifiers are trained taking into account the differences in availability of historical data (-60, -45, -30, -15 and 0 min) for each cell.

Table 2. Preliminary Extreme weather convective discrimination results.

History (min)	POD	FAR	F1
0, -15, -30, -45, -60	0.8	0.35	0.72
0, -15, -30, -45	0.77	0.36	0.7
0, -15, -30	0.8	0.4	0.67
0, -15	0.78	0.42	0.67
0	0.8	0.51	0.61

### 9.2 Time Results

Regarding the ship detection scenario, we proposed two target areas to process them in a configurable multi-board scheme. The first one would be the min strip scenario, covering 100 km<sup>2</sup> and processing 8400px· 12000px with 1 m/px of GSD. Table 3 shows the time elapsed by each ship detection algorithm in a dual-board scheme formed by the CEDH and an only one processing board. Assuming transfer delays and management task, it is possible to have ready the **products to be sent in 38 seconds, less than 1 minute**.

Table 3. Ship detection algorithms elapsed time.

	Time (s)
Calibration & Denoising	~17
Geolocation	~0
Land Mask Extraction	~4
Candidate Extraction	~12
Ship Discrimination + Alert Generation	~2
<b>Total Elapsed Time</b>	<b>~35</b>

The second target area is 1000 km<sup>2</sup> in a quad-board scheme, composed by the data handling board (CEDH) and 3 processing boards. As each processing board covers 100 km<sup>2</sup>, it is possible to process 300 km<sup>2</sup> in parallel and 3x 8400px x 12000px. With this distribution, an area of 900 km<sup>2</sup> with 3x 3x 8400px x 12000px can be estimated to be processed in 115 sec. Finally an area of 1200 km<sup>2</sup> with 4x 3x 8400px x 12000px can be estimated to be processed in

154 sec. Regarding the extreme weather scenario, it is processed in a dual-board scheme with only one processing board. The images processed are 1192px x 639px with 3 km/px of GSD, covering 6.855.192 km<sup>2</sup>. Only the project generation latency is tested for this scenario (i.e. L1 generation is not here estimated). Table 4 shows the time elapsed by each extreme weather algorithm in a dual-board scheme formed by the CEDH and an only one processing board. Assuming transfer delays and management task, it is possible to have ready the products to be sent in 15 seconds.

Table 4. Extreme weather algorithms elapsed time.

	Time (s)
Candidate Identification and Tracking	~8
Discrimination	~5
<b>Total Elapsed Time</b>	<b>~13</b>

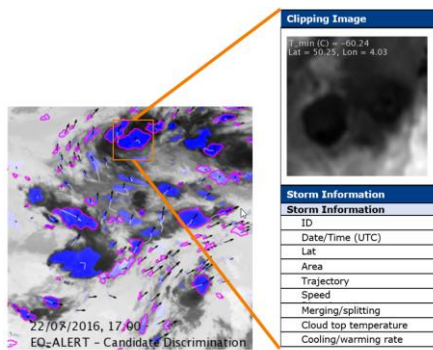


Figure 7. Extreme weather convective storm alert.

### 9.3 Products Examples

Figure 7 shows an example of an alert generated for a convective storm detected by the EO-ALERT Extreme Weather product. The alert messages contain comprehensive characterisation details like location, area, previous and predicted movement and temperature evolution for each convective cell, as well as a thumbnail image, which can be further evaluated on ground. An example of the alert product generated for the Ship Scenario is presented in Figure 8. The alert contains information about the ship such as length, position, and the confidence of detection, as well as a thumbnail image for visual assessment is included.

## 10. CONCLUSIONS

The proposed next generation satellite processing chain for rapid civil alerts, exploiting on-board processing, provides suitable performances in HW for image generation and processing, to support the conclusion that this architecture is the correct tendency and a promising solution to decrease product latency. In addition, the kind of COTS devices

proposed are increasingly used in space projects and contribute to put more computational power on-board than ever before. The rapid prototyping allows developing from a global perspective. Is important to note that the time requirements have been validated in HW. In the algorithm part, EO-ALERT Optical processing chain is focused on Ship Detection and Extreme Weather but can be used for many other applications which use ML and AI. With these techniques (AI and ML), normally used on ground, it is possible to send results to the end user using near real time links with global coverage. As a future work, additional functions could be implemented in the FPGA in order to increase the performance.

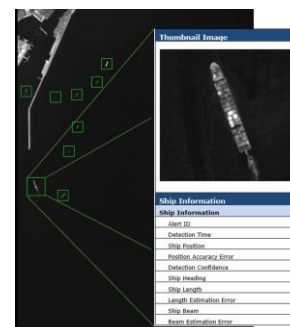


Figure 8. Ship detection alert.

## ACKNOWLEDGEMENTS

This project has been supported by the European Union and the H2020 Research and Innovation programme. We thank and acknowledge our partners DLR, OHB Italy, POLITO, TU-GRAZ, DEIMOS Imaging, for their collaboration and effort to achieve the project objectives.

## REFERENCES

- [1] Skauen, A. N., Berit Jahnsen, Tore Smestad et al. "NorSat-3 – Next Generation Norwegian Maritime Surveillance." (2019).
- [2] Otsu, N. (1979). A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics, 9(1), 62-66.
- [3] Wu, K., Otoo, E., Suzuki, K. (2009). Optimizing two-pass connected-component labeling algorithms. Pattern Analysis and Applications, 12(2), 117-135
- [4] E. Normand, "Single-event effects in avionics," in IEEE Transactions on Nuclear Science, vol. 43, no. 2, pp. 461-474, April 1996, doi: 10.1109/23.490893
- [5] R. W. Hamming, "Error detecting and error correcting codes," in The Bell System Technical Journal, vol. 29, no. 2, pp. 147-160, April 1950
- [6] PG187 UltraScale Architecture Soft Error Mitigation Controller v3.1, Apr. 2016, [online]