# D5.5

# Drone Sensors and In-Situ Sensors

| | |
|---|---|
| **Instrument** | Collaborative Project |
| **Call / Topic** | H2020-SEC-2016-2017/H2020-SEC-2016-2017-1 |
| **Project Title** | Multi-Hazard Cooperative Management Tool for Data Exchange, Response Planning and Scenario Building |
| **Project Number** | 740689 |
| **Project Acronym** | HEIMDALL |
| **Project Start Date** | 01/05/2017 |
| **Project Duration** | 42 months |
| **Contributing WP** | WP 5 |
| **Dissemination Level** | PU |
| **Contractual Delivery Date** | M22 |
| **Actual Delivery Date** | 04/08/2020 |
| **Editor** | Guido Luzi (CTTC) |
| **Contributors** | Juan Marchal Gomez (DLR), José Becerra Rivera (ICGC), Guido Luzi (CTTC) |

| Document History | | | |
|---|---|---|---|
| Version | Date | Modifications | Source |
| 0.1 | 27.03.2020 | First draft | DLR |
| 0.2 | 17.06.2020 | QA Ready | CTTC |
| 0.3 | 29.06.2020 | QA Done | PCF |
| 0.4 | 20.07.2020 | Request for revision | DLR |
| 0.5 | 29.07.2020 | Revised | CTTC, ICGC, DLR |
| 1.0.D | 30.07.2020 | Final version | CTTC |
| 1.0.F | 04.08.2020 | Version for approval | DLR |

# Table of Contents

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| API | Application Programming Interfaces |
| AVA | Avanti Communications LTD |
| AWS | Automatic Weather Station |
| CTTC | Centre Tecnològic de Telecomunicacions de Catalunya |
| DB | Database |
| DLR | Deutsches Zentrum für Luft- und Raumfahrt e.V. |
| GB-SAR | Ground Base Synthetic Aperture Radar |
| GUI | Graphical User Interface |
| ICGC | Institut Cartogràfic i Geològic de Catalunya |
| LOS | Line Of Sight |
| MAV | Micro Aerial Vehicle |
| RKT | Real time Cinematics |
| ROI | Region of Interest |
| POI | Point of Interest |
| ROS | Robot Operating System |
| RPi | Raspberry Pi |
| SP | Service Platform |
| SW | Software |
| ToC | Table of Contents |
| UAV | Unmanned Aerial Vehicle |

Intentionally blank

# Executive Summary

This document provides the update of the report D5.4, devoted to the technical specifications associated to drones and in-situ sensors, which are two of the HEIMDALL system inputs. Drones and in-situ sensors are connected to the HEIMDALL service platform. Information from these sensors contributes to facilitate end users' decision making.

In D5.4 the major focus was on the landslide in-situ sensor since they have been already integrated in the overall HEIMDALL system for Releases B due at that time. Since then, and describing the major updates in this document, the focus of the integration work has been on the drones system. Compared to the previous issue, the development of the base station GUI has been finished, hotspots detection algorithms and routing algorithms were developed and implemented, and the drone system was tested and debugged with the complete system in both simulation and hardware experiments. Furthermore, the drone's payload has been integrated and the system was interfaced to the HEIMDALL service platform. The updates can be found in section 4 specification. For the developments of the algorithms and for testing them, a hotspot measuring campaign has been performed in Catalonia.

The in-situ sensors module was delivered for Release B, since then no new features have been integrated but maintenance and updating of the Barberà de la Conca's network and its communication system with the SP have been performed.

# 1 Introduction

This document describes the work done as part of task 5.2 (Drone Sensors and In-Situ Sensors). Task 5.2 provides the means for in-situ monitoring and early detection of two hazard types: forest fire and landslides. The work in this task uses the sensor technologies already described in D5.4, i.e.:

- Swarm of Micro Aerial Vehicles (MAVs) for forest fire which includes the following subtasks:
  - Development of a MAVs swarm for forest fire hotspot detection, in-situ monitoring and surveillance with thermal sensors.
  - Equipage of a swarm with thermal cameras, a microcomputer for on-line processing of acquired data, and communication and navigation modules.
  - Development of a portable control station, whose prime goals include the definition of regions of interest (ROIs) through an operator and communication of relevant sensor data/decisions to a central location.
  - Development of an algorithm for an optimal ROI tessellation and organization of autonomous swarm movements on a constructed grid to minimize the exploration time, while at the same time avoiding collisions between MAVs.
  - Development of an algorithm for online processing of sensor data for fast identification of potential hot spots.
  - Execution of flight tests to validate the specifications of the integrated system.
- Geotechnical/hydrological sensors and displacement remote sensors for terrain movement in anticipation of landslides. First group refers to sensors that are installed in contact with the monitored target (inside or attached to the moving terrain) and the second group corresponds to sensors that measure the surface changes by measuring distances from a fixed point; in the case at hand, a ground-based synthetic aperture radar (GB-SAR). Task 5.2 includes the following subtasks:
  - Installing in-situ sensors (crackmeters, tiltmeters, piezometers, …) in an area identified as prone to slide or with an ongoing landslide, allows hazard managers to raise warnings when these sensors measure an increment on landslide speed. So that, an In-Situ Sensors module has been develop to integrate these measures in the HEIMDALL platform.
  - Provision three-dimensional (multi-temporal) maps of the deformation field allowing detecting the deformation of the whole unstable area and hence minimizing the errors in reconstruct the landslide activity starting from punctual information of deformation. In this context, the tasks to be done are: (i) continuous time series of the deformation, which can provide information on the temporal evolution of the displacement; and (ii) multi-temporal maps of the displacement affecting the slope in time.
  - Real-time monitoring: this task consists of performing a few months real time-monitoring campaign of a real test site, including data acquisition and processing. The main goal is to provide the temporal evolution of the monitored deformation. This task provides a demonstration example of how GB-SAR can provide key inputs for landslide emergency management.
  - Long term monitoring: this task will consist in the monitoring of a site during the whole or part of the project, depending on the site conditions. The task will include the data acquisition and processing in order to provide the temporal evolution of the monitored deformation. The main aim of this task it to provide an example of the use of GB-SAR derived maps during landslide risk prevention phase in places where the satellite data is not available or its revisiting time is not adequate to the observed phenomena dynamics.
  - In areas with higher susceptibility it is necessary to measure in real time to provide evolution of terrain deformation. GB-SAR data will be complemented by other technics as geodesic or topographic surveys in order to obtain realistic scenarios and simulations for an accurate response planning. Special

relevance has the incorporation of subsurface data (using geotechnical instrumentations: extensometers, piezometric …) as warning factor.

Since each of the sensor technologies has different requirements and employ different technologies, they are described separately in this document.

This document is organised as follows:

- First, we specify our technical requirements in Section 2.
- Section 3 provides a description of the technical requirements and illustrates how our sensor technologies interact with the HEIMDALL system.
- In Section 4, we describe our system module functionality at a high level, highlighting the blocks that compose our system.
- In Section 5 we extend the previous high-level explanation of the system and explain in detail the individual sub-blocks that compose our system.
- Section 6 presents the test plan to validate our system architecture and associated functionality. This is followed by conclusions in Section 7.

# 2 Technical Requirements

In this section, we specify the technical requirements of drones and in-situ sensors. The requirements are presented according to the sensor technology being considered. Then, for each sensor technology, we specify: (i) interface requirements, (ii) functional technical requirements, and (iii) other requirements.

## 2.1 Swarm-Based Forest Fire Hotspots Detection

### 2.1.1 Interface Requirements

#### 2.1.1.1 Hardware Interfaces

The MAV sensor system is composed of three main elements (see Figure 2-1):

- A DronesGUI that allows interaction with HEIMDALL service platform, an operator, and MAVs.
- An autonomous MAV hot spot detection system that runs on a base station, termed as HotspotBase (blue block); and
- An autonomous MAV hot spot detection system that runs on the drones, termed as HotspotDrones (green block).

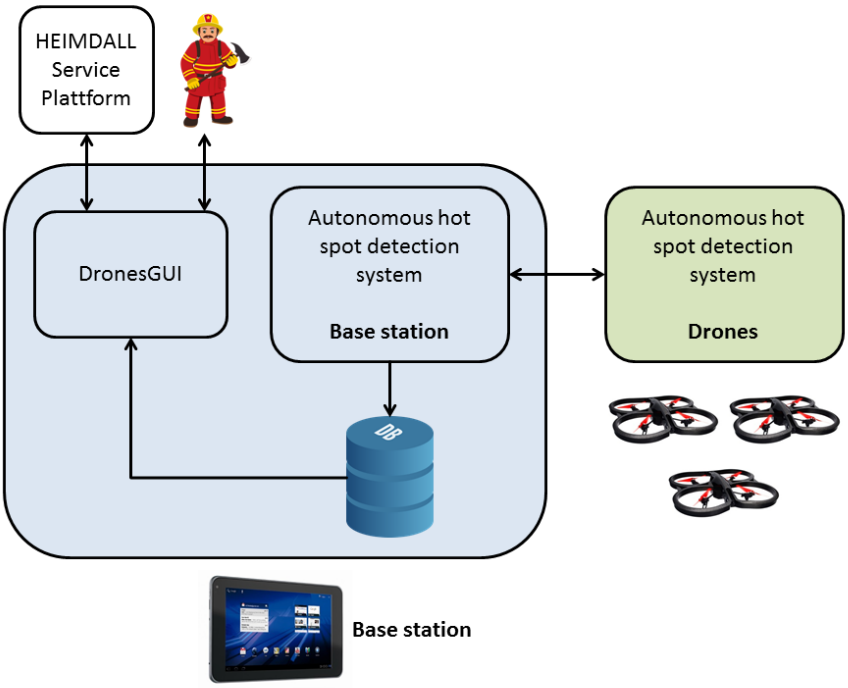


Figure 2-1: Swarm-based forest fire hotspots detection system interfaces

MAVs interact with the rest of the system developed in HEIMDALL through a DronesGUI that runs on a base station. Base station is the only hardware the end-user interacts with. Therefore, only base station hardware requirements are considered in this section. According to the user requirements, the Base station must:

- be portable so that an operator can take it to the field. Preferably it should be a tablet, as indicated by end users.
- have processing specifications higher than of a Google Pixel C tablet. That is:
  - CPU: Quad-core 1.9 GHz

    - RAM: 3 GB
- offer the operator the possibility to interact with it, either through a keyboard and mouse, or through a touch screen.

#### 2.1.1.2 Software Interfaces

The software interfaces between MAVs and the rest of the system can be further split into the following interfaces:

➢ **DronesGUI-HotspotBase interface**

The DronesGUI must transmit commands to HotspotBase in order to:

- start/stop the hotspot detection algorithm;
- request a real time video of a specific drone's view;
- select whether a detected hotspot should be monitored or not;
- define a region of interest (ROI) where we aim to search for hotspots. A ROI must be specified by a list of points that correspond to the ordered vertices of a polygon that defines the region;
- trigger a return to base procedure. This will force drones to stop the algorithm's execution and return to base.

These instructions are indirectly commanded through a database (DB).

➢ **DB-DronesGUI interface**

The database (DB) must provide a Python API that will allow the DronesGUI to retrieve the DB information.

➢ **HotspotBase-HotspotDrones interface**

The base station must run Ubuntu 16.04.

➢ **DronesGUI-HEIMDALL Service Platform interface**

The base station must run Ubuntu 16.04.

#### 2.1.1.3 Communication Interfaces

Communication interfaces are associated to the link between the DronesGUI and the HEIMDALL system: with the HEIMDALL service platform (described in deliverable 4.1). To this end, an API shall be provided by the HEIMDALL service platform to allow us to easily transmit information from base station to the HEIMDALL system through the DronesGUI. As communication link, we use the satellite-based internet link which is provided by AVA (described in deliverable D4.16).

### 2.1.2 Functional Technical Requirements

#### 2.1.2.1 Short Term Features

No short term requirements have been identified.

#### 2.1.2.2 Mid Term Features

Table 2-1: Functional technical requirement for drones - TR_DataSitu_6.

| | |
|---|---|
| Requirement ID: | TR_DataSitu_6 |
| Related SR(s): | • Sys_DataSitu_7 |
| Description: The MAVs system shall be able to define a ROI, where drones should search for hotspots. | |
| Rational: Drones need a region where they should look for hotspots. | |
| Stimulus: After a fire, there are remaining hotspots. Firefighters will trigger the system in areas where a fire took place. | |
| Response: An alarm message is sent when a hotspot is detected. | |
| Verification Criterion: Artificial hotspots at known locations will be ignited, and then drones should find them all. | |
| Notes: None | |

Table 2-2: Functional technical requirement for drones -TR_DataSitu_7.

| | |
|---|---|
| Requirement ID: | TR_DataSitu_7 |
| Related SR(s): | • Sys_DataSitu_7 |
| Description: The system shall be able to generate points of interest (POIs) within the ROI where drones should fly to and check whether there is a hotspot. | |
| Rational: As we aim for an autonomous system, this should automatically generate POIs where drones should fly to. | |
| Stimulus: Once a ROI is defined, POIs should be created. | |
| Response: Confirmation message that POIs were created. | |
| Verification Criterion: In simulations, artificial ROIs will be created and then we will check whether generated POIs cover the complete ROI. | |
| Notes: None | |

Table 2-3: Functional technical requirement for drones - TR_DataSitu_9.

| | |
|---|---|
| Requirement ID: | TR_DataSitu_9 |
| Related SR(s): | • Sys_DataSitu_7 |
| Description: Drones shall be able to take visual and thermal pictures, process them and decide whether there is a hotspot at a particular location. | |
| Rational: Visual and thermal pictures are state-of-the-art to detect fire hotspots. | |
| Stimulus: Once a drone reaches a commanded waypoint, pictures should be taken. | |
| Response: Drones decide whether there is a hotspot or not. | |
| Verification Criterion: Artificial hotspots will be created, and drones should be able to identify | |

| |
|---|
| them according to pictures. |
| Notes: None |

Table 2-4: Functional technical requirement for drones - TR_DataSitu_10.

| | |
|---|---|
| Requirement ID: | TR_DataSitu_10 |
| Related SR(s): | • Sys_Gui_85<br>• Sys_Gui_87<br>• Sys_Data_4<br>• Sys_DataSitu_5 |
| Description: Drones shall be able to send the following gathered information to the base station which will be stored in a DB:<br>• Drones: ID, position, timestamp.<br>• POIs: ID, position, drone assigned/visitor, thermal and RGB pictures (time and geo-referenced).<br>• Hotspots: ID, position, drone visitor, timestamp, temperature, thermal and RGB pictures. | |
| Rational: The operator should have updated information through the DronesGUI. | |
| Stimulus: Once new information is collected, this should be sent to the base station. | |
| Response: Confirmation message that information has arrived and was stored. | |
| Verification Criterion: Information is generated and correctly saved in the DB. | |
| Notes: None | |

Table 2-5: Functional technical requirement for drones -.TR_DataSitu_11.

| | |
|---|---|
| Requirement ID: | TR_DataSitu_11 |
| Related SR(s): | • Sys_DataSitu_3<br>• Sys_DataSitu_4<br>• Sys_DataSitu_7<br>• Sys_DataSitu_8 |
| Description: Drones shall be able to send a real time video of the current situation. | |
| Rational: Video is the best source of information for end users. | |
| Stimulus: Video will be continuously transferred while drones fly. In the DronesGUI, only video from one drone can be displayed, so a specific drone must be selected by the operator. | |
| Response: Video in GUI. | |
| Verification Criterion: We should be able to see a drone's video in the GUI. | |
| Notes: None | |

#### 2.1.2.3 Long Term Features

No long term features have been identified.

### 2.1.3 Other Requirements

No other requirements have been identified.

## 2.2 GB-SAR based Landslide Monitoring

### 2.2.1 Interface Requirements

#### 2.2.1.1 Hardware Interfaces

The GB-SAR measuring system is composed of a radar transceiver, a rail for sensor motion, controlled by software (SW) present in a laptop, and a powering system. The acquiring configuration is defined at the start of the campaign through an internal GUI and requires setting the following parameters:

- Range and azimuth resolution (in meters);
- Maximum Measuring range (in meters);
- Temporal interval between two consecutive images (in seconds).

The GB-SAR is installed in a stable site, i.e. in an area not affected by the deformation which is mechanically steady. The observation geometry is selected to estimate the main component of the expected displacement/deformation, and usually no more than 3 km far from the monitored terrain surface. The installation can be performed in one hour, provided that a structure to install the apparatus (usually concrete base) is already available. In the case of continuous monitoring, the system, through the laptop, is remotely controlled using a wired or wireless link to the web. In the discontinuous case, one day long campaigns are carried out, and the system provides a set of twenty images per campaign. In this case a direct link is not necessary to provide data to HEIMDALL service platform.

The main hardware components can be categorised as mechanical and electronic.

- Mechanical:
    - A metal rail.
    - Stable basement provided with fast and robust interface: solid metal threaded rods to fix the rail.

- Electronic
    - Power unit, including two batteries.
    - A laptop, including a wireless link unit.
    - A video camera to monitor the area.

For the continuous monitoring case, the hardware interface between the system and the service platform (SP) is a server at CTTC which includes a standard web receiving link, connected with the system, whose main characteristics are:

- CPU: > 2.2 GHz;
- RAM: > 8 Gigabyte;
- Hard Disk: 2 Terabyte memory capacity assures 12 months of continuous acquisition.

GB-SAR hardware is configured on site through the laptop and interacts with the rest of the system developed in HEIMDALL through the link station which transfers data to the SP.

In the discontinuous case, the hardware interface is a memory stick where data acquired during the daily campaign are stored. Figure 2.2 shows a drawing of the GB-SAR system.

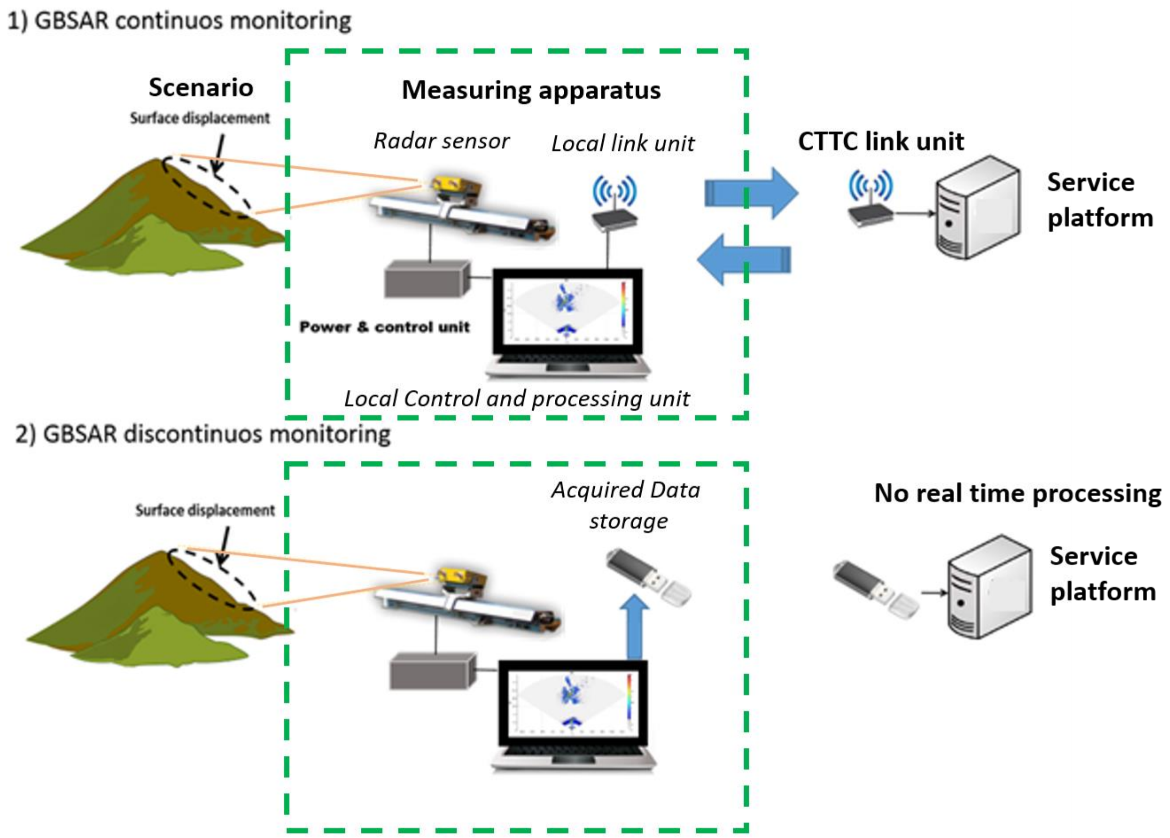


Figure 2-2: Hardware and communications configurations for GB-SAR acquisitions.

#### 2.2.1.2 Software Interfaces

For the GB-SAR outputs, the SW interface consists of a tool common to both configurations, continuous and discontinuous. It allows the setting of the acquisition parameters and provides SAR images in a quasi-real time (10 minutes step) and related updated deformation maps. Different systems are available from the market [1]. The apparatus used in HEIMDALL is an IBIS-L and the stored data are processed with a SW tool available from CTTC [2].

The software interfaces between the GB-SAR and the rest of the monitoring system is a GB-SAR GUI located in the laptop (Windows 7) The GUI allows selecting a specific area, and to upload the two main products: the deformation maps and temporal series from the DB.

#### 2.2.1.3 Communication Interfaces

The communication interface between the GB-SAR and the SP, in the case of continuous monitoring, consists of a wired or wireless link, remotely controlled and managed by the laptop controlling the measuring system. A standard tool, e.g. remote desktop/Teamviewer® is used. The data throughput and storage depend on the applied configuration: a record (image) of about 10 Megabyte every 5 minutes is delivered in the continuous case. A connection of 7 Mbit/s fits throughput requirements. In the discontinuous case, the system provides a set of twenty images per campaign. In this case the remote control is not necessary, and images can be stored in the laptop's internal memory and easily transferred through a portable 8 Gigabyte memory.

### 2.2.2 Functional Technical Requirements

#### 2.2.2.1 Short Term Features

No short term requirements have been identified.

#### 2.2.2.2 Mid Term Features

Table 2-6: GB-SAR functional technical requirements - TR_DataSituMon_1.

| | |
|---|---|
| Requirement ID: | TR_DataSituMon_1 |
| Related SR(s): | • Sys_DataSitu_1 |
| Description: The system shall store the new incoming data from the GB-SAR sensor in the Service Platform. | |
| Rational: Provide a deformation map every ten minutes will be key in the situation assessment for the landslide case. | |
| Stimulus: New data from the GB-SAR are available. | |
| Response: Data are provided to the SP. | |
| Verification Criterion: We are able to see the deformation map. | |
| Notes: None | |

Table 2-7: GB-SAR functional technical requirements. TR_DataSituMon_2.

| | |
|---|---|
| Requirement ID: | TR_DataSituMon_2 |
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:**<br>The system shall be able to operate with original variables from the GB-SAR. | |
| Rational: Starting from line-of-sight displacement estimates provided by the GB-SAR, other parameters can be necessary to the users (e.g. velocity, acceleration). | |
| Stimulus: Formula is introduced in the software by the developers. | |
| Response: New variables are included in the database. | |
| Verification Criterion: New data are available: the total number of variables increases. | |
| Notes: None | |

#### 2.2.2.3 Long Term Features

No long-term requirements have been identified.

### 2.2.3 Other Requirements

No other requirements have been identified.

## 2.3 Geotechnical/Hydrological Landslide Monitoring

### 2.3.1 Interface Requirements

#### 2.3.1.1 Hardware Interfaces

The geotechnical/hydrological sensors for landslide monitoring are connected to loggers, which acquire, store and send the geotechnical/hydrological/displacement data. Data is initially sent to an internal ICGC server. HEIMDALL System gets the data from the ICGC server through HTTP, through a REST API called by the Service Platform (Figure 2-3). There, data from the sensors and derived information will be shown to the user through the GUI. Based on the features that will be shown from the sensors, the Service Platform does not need any special requirements in terms of hardware.

As an example of hardware system requirements, more advanced commercial sensors data management software (with much more specific technical features) have the following system requirements:

- CPU: 1.5 GHz or faster. 2.2 GHz or faster is preferable.
- RAM: 4 Gigabyte or more.
- Hard Disk: 4 Mbyte/year of hard disk space considering 10 different sensor variables and 15 minutes sample time.

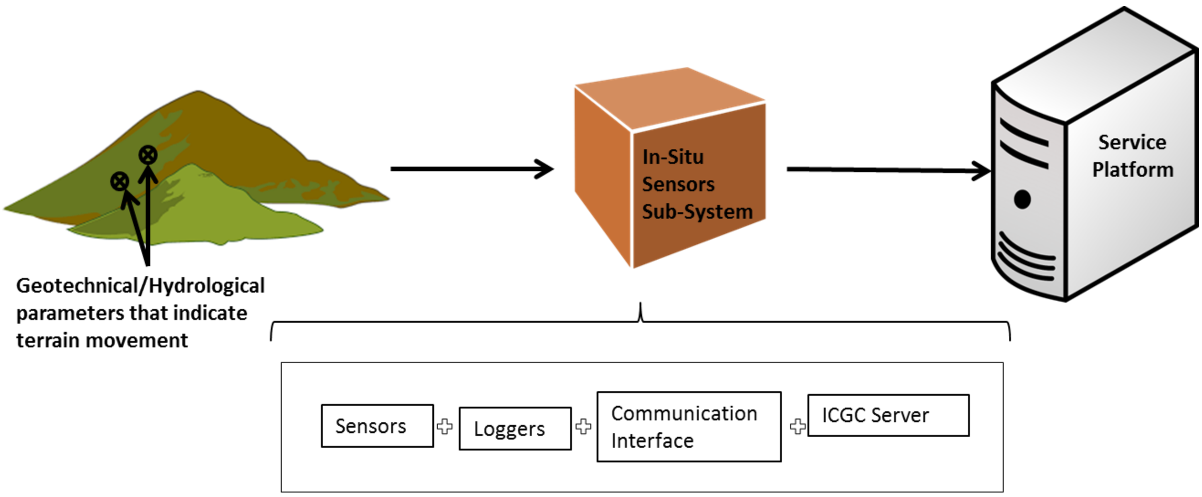


Figure 2-3: Process of data acquisition from the monitored area until it reaches the HEIMDALL System (Service Platform). Components of the module are indicated.

#### 2.3.1.2 Software Interfaces

Data from sensors and some derived parameters (for example raw data translated into engineering units) will be available in the databases of an ICGC server. This data can be automatically obtained by the Service Platform (SP) in JSON format. Once data is available in the SP, it will be displayed through the GUI in plots of raw data variables and engineering units variables (both considered as DB variables).

#### 2.3.1.3 Communication Interfaces

As mentioned above, data from sensors will be available at the ICGC Server, to be obtained by the Service Platform in JSON format. Data from the sensors will be transferred to the server at different intervals depending on the sensors and the test site. It will be either daily or twice/day and not in real time.

Since transfer will be through HTTP, it is not needed to specify any synchronization mechanism or data link capacity. Visualization of the data will be through the HEIMDALL GUI.

### 2.3.2 Functional Technical Requirements

#### 2.3.2.1 Short Term Features

No short term requirements have been identified.

#### 2.3.2.2 Mid Term Features

Table 2-8: Geotechnical/hydrological monitor functional technical requirements. TR_DataSituGeo_1.

| Requirement ID: | TR_DataSituGeo_1 |
|---|---|
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:** The in-situ sensors module shall provide a DB with the historical and new incoming data from the in-situ sensors. | |
| Rational: In-situ sensors will be key in the situation assessment for the landslide case. | |
| Stimulus: New data available requested by SP. | |
| Response: Data are added to the database. | |
| Verification Criterion: Database is updated. | |
| Notes: none | |

Table 2-9: Geotechnical/hydrological monitor functional technical requirements. TR_DataSituGeo_2.

| Requirement ID: | TR_DataSituGeo_2 |
|---|---|
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:** The in-situ sensors module shall be able to provide not only original raw data from the sensors, but also data derived from operations of original data. These data should be provided as a DB, too. | |
| Rational: Sometimes data directly extracted from the sensors does not provide the relevant information to understand the physical process. For example, data from sensors (mV, A, ohms, Hz, etc.) must be transformed into engineering units (kPa, cm, %, etc.), in order to make it understandable for users. | |

| |
|---|
| Stimulus: Formulas to transform raw data are introduced by the developers. |
| Response: Transformed data is added to the DB. |
| Verification Criterion: Total number of variables increases. |
| Notes: None |

Table 2-10: Geotechnical/hydrological monitor functional technical requirements. TR_DataSituGeo_3.

| Requirement ID: | TR_DataSituGeo_3 |
|---|---|
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:** The in-situ sensors module shall provide time series of data (raw and virtual variables) to be plotted by the GUI. | |
| Rational: Graphic visualization of data through plots is more useful to identify trends, and to rapidly see the evolution of the parameters. | |
| Stimulus: Every time a new record is included in the DB, it is visualized graphically in plots that include this variable. | |
| Response: Data is added to the plots. | |
| Verification Criterion: Message indicating last update of the plot. | |
| Notes: None | |

### 2.3.2.3 Long Term Features

Table 2-11: Geotechnical/hydrological monitor functional technical requirements. TR_DataSituGeo_4.

| Requirement ID: | TR_DataSituGeo_4 |
|---|---|
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:** The in-situ sensors module shall provide additional information of data trends, to be used to build automatic reports to monitor the status of the terrain. | |
| Rational: When a scenario is open and in continuous surveillance, it is useful for the stakeholders to receive updated information without the necessity of having to log into the platform and check the situation. | |
| Stimulus: Automatic report creation is pre-configured to be created at a certain frequency by developers | |
| Response: Report is created and made available to the platform. | |
| Verification Criterion: A “log message” appears in the system saying that the automatic report has been sent. | |
| Notes: None | |

Table 2-12: Geotechnical/hydrological monitor functional technical requirements. TR_DataSituGeo_5.

| | |
|---|---|
| Requirement ID: | TR_DataSituGeo_5 |
| Related SR(s): | • Sys_DataSitu_1 |
| **Description:**<br>The in-situ sensors module should be able to send alert messages in case of exceedance of pre-defined thresholds | |
| Rational: When an indicator of terrain movement exceeds a certain threshold, it is required that stakeholders receive this information in order to take actions as soon as possible. | |
| Stimulus: An indicator of terrain movement exceeds a certain pre-defined threshold. | |
| Response: An alert message is sent to pre-defined stakeholders. | |
| Verification Criterion: A “log message” appears in the system saying that the alert message has been sent. | |
| Notes: None | |

### 2.3.3 Other Requirements

No other requirements have been identified.

# 3 Reference Architecture

Drones and in-situ sensors constitute two of the system inputs to HEIMDALL service platform, as illustrated in Figure 3-1.

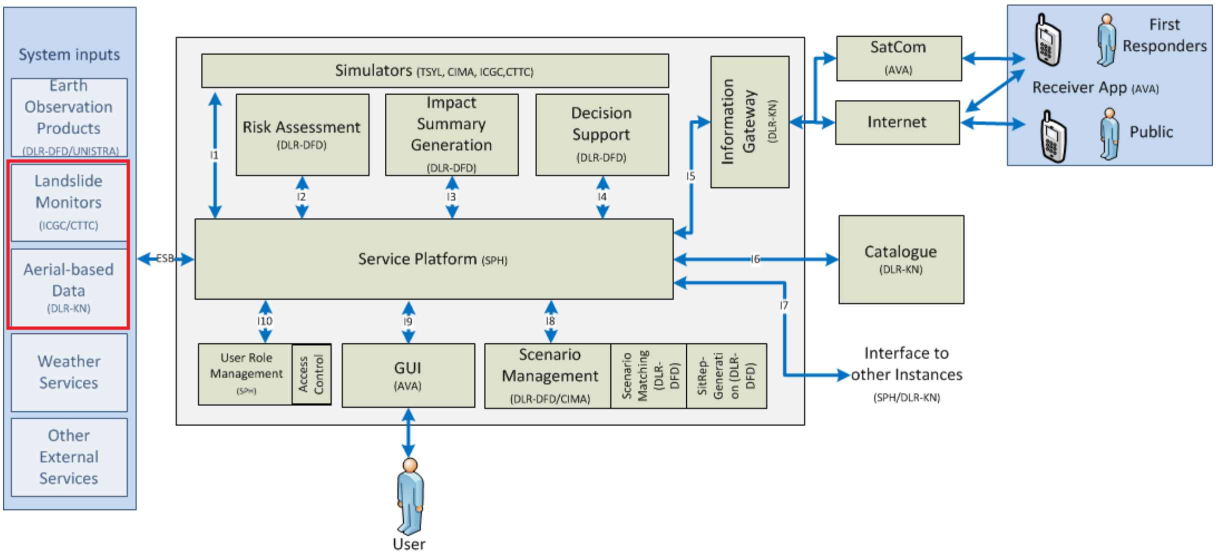


Figure 3-1: Drones and in-situ sensors within HEIMDALL overall architecture.

We describe the interaction and interfaces between the drones and in-situ sensors, and the HEIMDALL service platform below.

## 3.1 Swarm-Based Forest Fire Hotspots Detection

HEIMDALL offers an aerial-based data product to support firefighters. In particular, our product allows to monitor a ROI and detect hotspots using a swarm of drones. The use of a swarm of drones, in contrast to a single drone, permits accelerating the monitoring process.

As we previously mentioned, data from drones constitute an input to the service platform. Specifically, we transmit the following information from drones: drones' position, RGB and thermal pictures taken by drones, and real time video from drones. To transmit this information from drones, we use an API provided by the SP. Through this API, we can transmit information from the operator's tablet to the HEIMDALL service platform. As the communication channel to transmit such information, we use a satellite link. The use of a satellite link is important as, typically, forest fires take place in regions with limited signal coverage. Therefore, the use of a satellite link will allow an operator to transmit information from remote locations directly to HEIMDALL service platform. Note that we designed our system such that other communication links like e.g. 4G could be used to transmit drone's information.

## 3.2 Landslide monitors

HEIMDALL platform will be equipped with a module for landslide monitoring, with a goal to monitor the terrain stability conditions of an area that has been identified as unstable, by installing several sensors in-situ. These sensors will measure parameters that are directly or indirectly related to the stability of the terrain. Monitoring slopes is a key tool to better understand the kinematic aspects that may cause instabilities and failures [3]. Therefore, data from sensors can provide relevant information to predict failures in advance and act as warning systems.

In order to decide which sensors and where to install them, the user should have an idea of the extension of the unstable area, the direction of the expected movement, as well as of the mechanism of instability. Once the monitoring system is installed, this module will not need other inputs, since the measurements of the sensors are the input data to the module. By the analysis of this data, four main products will be obtained as output. Firstly, post-processed data from terrestrial radar. Secondly, data from the geotechnical/hydrological sensors which will provide information on the magnitude of the terrain movement or parameters that affect the instability. Thirdly, results of geodesic and topographic surveys and finally, the data available, which may come from sensors, radar or surveys will be summarized in a report.

The landslide monitoring module will interact with the HEIMDALL Service Platform through a REST API. The data collected by the sensors will be sent to the Service Platform and through the GUI these data will be available to the user.

# 4 Module Functionality

In this section we describe the different building blocks within the sensors under consideration. Here we introduce a high-level description of the different blocks, together with the corresponding internal interfaces.

## 4.1 Swarm-Based Forest Fire Hotspots Detection

Aerial sensors have increased its popularity in recent years [4], [[4]. In particular, most used aerial platforms for civil applications are the so-called Micro Aerial Vehicles (MAVs). In the scientific and industrial communities, the use of a single MAV has been widely studied and exploited [5]-[12]. In contrast, the simultaneous use of multiple MAVs, instead of one, offers clear advantages. In particular, it offers advantages in terms of efficiency to gather information, and robustness [13]. However, the use of multiple MAVs introduces additional challenges as the system's complexity increases. In HEIMDALL, we developed novel algorithms to cope with such challenges, which will allow us to monitor a hazard with multiple MAVs that fly simultaneously.

Main task of aerial sensors within HEIMDALL is to allow an operator to monitor a hazard. In particular, we consider hotspots that may remain after a fire in a local area. To monitor local areas, multi-rotor MAVs have been proven to be the most effective type of aerial sensors [14]. Multirotor MAVs have two main advantages over fixed wing MAVS. First, they can hover, which is a fundamental characteristic in case an operator intends to monitor a specific location continuously from the same perspective. Second, they have better manageability than their fixed wing counterparts.

Our goal in HEIMDALL is to develop algorithms that will allow multi-rotor MAVs to identify potential hotspots. Hotspots may have different shapes and sizes. To guarantee that even small hotspots will be identified, we use state-of-the-art thermal sensors. Such sensors typically have a weight of a few hundred grams. As thermal sensors need to be carried by the MAVs, we decided to use a platform that admits a high weight payload. Note that in addition to the thermal sensor, another sensors and computers should be carried by the drone. In particular, we use MAVs that are based on DJI S900 frame. Note that in HEIMDALL we develop software, not hardware. This implies that the software developed in HEIMDALL could be easily transferred to fixed-wing UAVs.

In the literature, there does not exist any system ready-to-use that allows a swarm of drones to autonomously detect forest fires hotspots. We can find some work like [15]. However, these works are based on simulation and do not consider crucial aspects, like how to deal with inter-drones collisions, or how to cover large territories. Moreover, work in the literature does not propose a system architecture that allows us to develop a swarm hotspot detection system.

Therefore, to the best of our knowledge, we can state that in HEIMDALL we develop a hotspot detection system that is ready-to-use by end users. In next sections we describe our system components and architecture.

Our system is composed of two main components:

- Base station: it is the core of the system. It allows the interaction between a user and the drones. This is realized through a tablet, with which the user can send commands to the drones and see the information gathered by the drones (e.g. thermal and visual images). The tablet runs two main modules. First, a graphical user interface that allows user interaction. Second, a database that acts as a bridge between the users and the drones.

- Drones: these are sensors platforms that are equipped with a thermal and a visual camera. Task of the drones is to scan the area autonomously to search and identify potential hotspots. Drones also have a copy of the database to store information that cannot be directly transmitted to the base station.

Figure 4-1 shows a block diagram that depicts the interaction between the main components of our system.

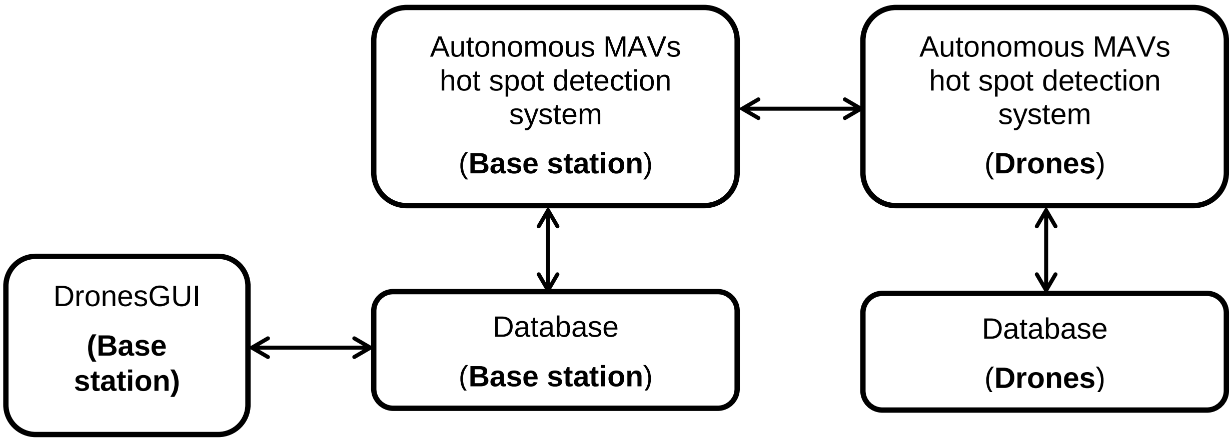


Figure 4-1: Block diagram from modules of swarm-based forest fire hotspots detection system.

## 4.2 GB-SAR-Based Landslide Monitoring

A portable, Ground Based SAR (GB-SAR) installed in a stable area in front of a landslide can provide radar images, using as working principle that used by space borne radar, but moving the microwave transceiver along a linear rail. Ground-based radar installations are apt for monitoring smaller regions like small urban area or single hillsides. As for satellite cases, GB-SAR radar images acquired at different times can be processed by means of interferometric techniques, when the decorrelation among different images is maintained low. The displacement obtained through the interferometric processing of SAR images, is the only the LOS component of the entire displacement. In ground-based observations, there is the necessity of finding a site with good visibility and from where the component of the expected displacement along the LOS is the major part of the monitored deformation. Several studies demonstrated the effectiveness of GB-SAR for remote monitoring of terrain slopes and as an early warning system to assess the risk of rapid landslides.

The main components of a GB SAR system are:

- A radar system.
- A power unit usually supported by batteries.
- A control and processing unit, a laptop, which provides the radar images in real time.
- A link between the processing unit and the receiving portal.
- A receiving portal.
- A database where data are available to the users.

In Figure 4-1 a simplified scheme of a GB-SAR system is shown.

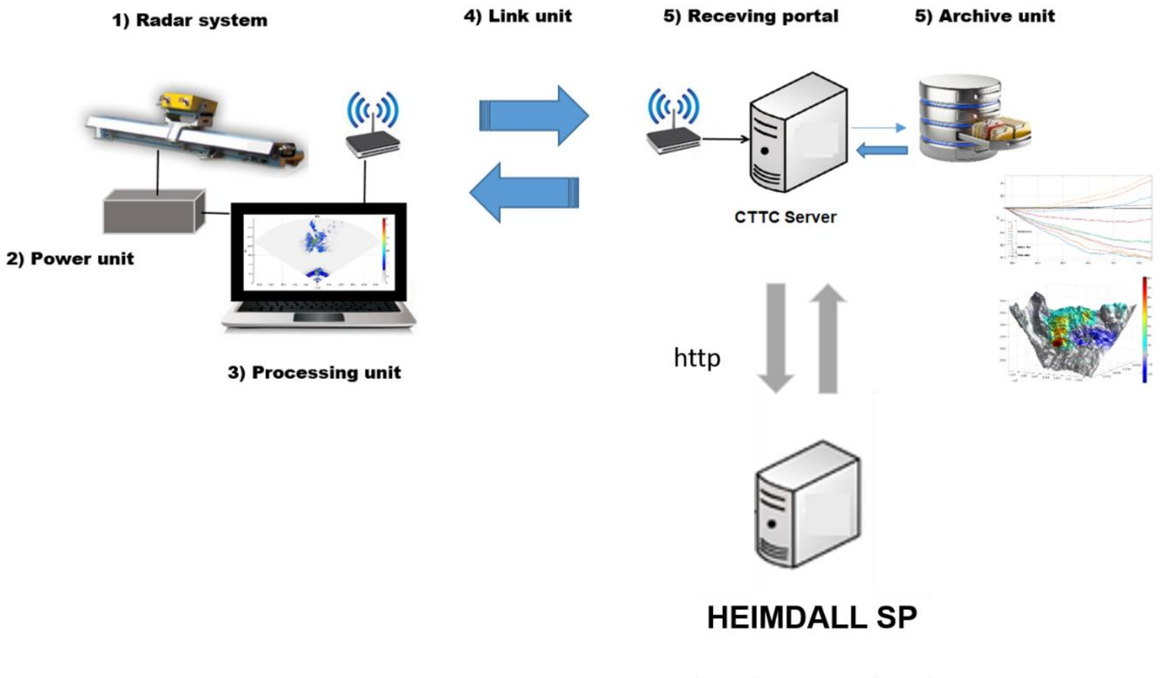


Figure 4-1: Scheme of the hardware and connection among the sub system of GB-SAR acquisitions.

## 4.3 Geotechnical/Hydrological Landslide Monitoring

Geotechnical and hydrological sensors to monitor landslides are connected to loggers, which acquire, store and send the geotechnical/hydrological/displacement data. Monitoring has two main objectives: a) scientific, to gather information about the instability mechanisms, responsible for the failure as well as the propagation of moving masses, and b) social, as early warning and alarm systems, to mitigate the associated risk to these phenomena [17], [18].

In HEIMDALL system, data from the sensors is sent to an internal server (ICGC) and, from where it is then periodically obtained by the Service Platform (Figure 4-2).

The geotechnical/hydrological landslide monitoring module is composed of 4 submodules:

- Sensors;
- Loggers;
- Communication interface;
- Server.

Details of these sub-modules can be found in Section 5.3.2. Data from geotechnical/hydrological parameters of the terrain (directly related to stability and terrain movement) is measured using sensors, which are installed in-situ, in areas where instabilities have been identified. These data are stored in the loggers, which are physically connected to the sensors, and are programmed in order to obtain data from them. Loggers provide the power that sensors require in order to take measurements. Data is sent from the loggers to the server at pre-defined time intervals, through a communication interface. Several types of communication interfaces exist and depending on the features of the site one or several of them are selected. As an example, in areas with great distances between sensors it may be more useful to use a wireless system rather than a wired system. However, in areas where sensors are placed next to each other it may be useful to use a wired system. Once in the server, data is retrieved by the Service Platform through a REST API.

The HEIMDALL System will receive the data from the ICGC server through an API at the Service Platform. There, sensors data is provided to the user through the user interface in order to inform about the changes in geotechnical/hydrological parameters of the terrain that may indicate terrain movement. Figure 4-2 shows the process of data acquisition where the communication protocol between server and SP and data visualization system are indicated

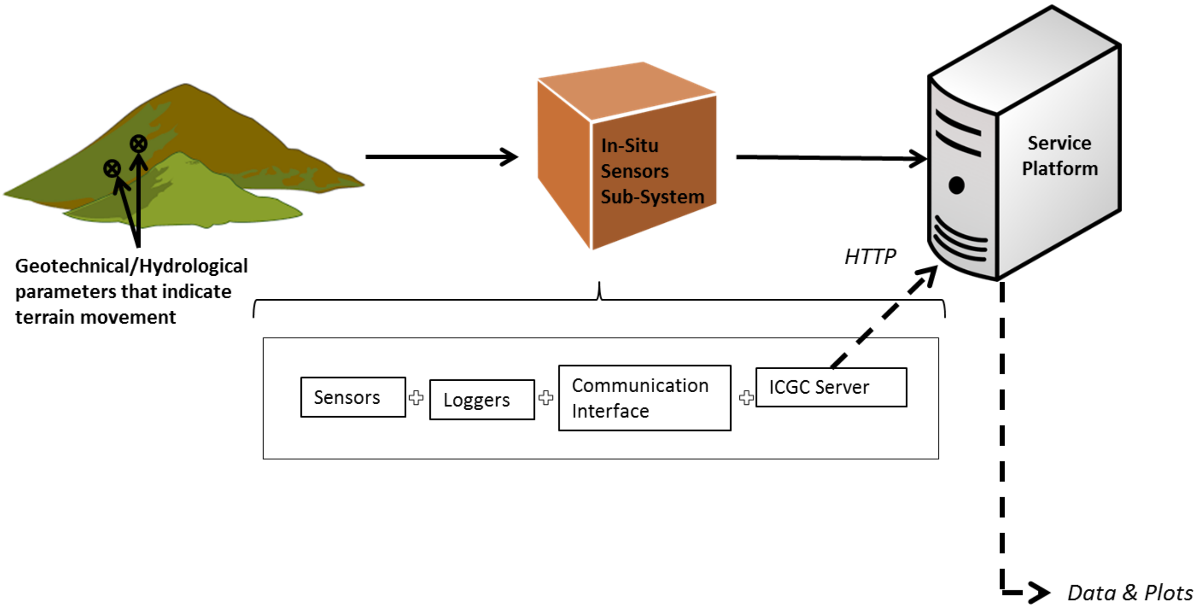


Figure 4-2: Process of data acquisition from the monitored area until it reaches the HEIMDALL System (Service Platform).

# 5 Technical Specification

In this section we describe in detail the different sub-blocks that compose our system. For each of the considered sensors, a system overview is presented first, followed by a detailed description.

## 5.1 Swarm-Based Forest Fire Hotspots Detection

### 5.1.1 System Overview

Our proposed system consists of two main components: a base station, and drones. Therefore, we detail the technical specifications for these two components separately. For both the base station and the drones, we separate modules in hardware and software modules. This way, it is easier to understand how we conceived the system.

#### 5.1.1.1 Base Station

Base station has two main functionalities. First, it allows an operator to control the system with a tablet. Second, it provides the entire communication infrastructure required to communicate with the drones and with the HEIMDALL service platform.

Next we detail the corresponding hardware and software modules that compose the base station.

##### *5.1.1.1.1 Hardware modules*

In an emergency situation, a base station needs to be easy to transport and deploy in order to improve the response time. We address this situation by using very compact equipment, consisting of two tripods, one case and a tablet. The user only needs to place the tripods, connect the GPS antenna and turn on the devices.

Figure 5-1 shows the block diagram of the base station. The RTK base station only interacts with the UAVs. It consists of a tripod with a GPS antenna that is connected to a case that contains a Swift Navigation Piksi Multi system [19] and a 12 volt battery. It also includes a long range radio link to transmit the GPS corrections to the different UAVs.

The UAVs are expected to fly distances outside the range of a conventional WiFi network, which is usually about 100 meters. If a hotspot is detected, the end user may need visual feedback to clarify if it is a real problem or not. We extend the range of a conventional WiFi router by using the long range communication device Mobilicom MCU-30 [20]. This device can operate at a programmable frequency range of 2.3-2.7 GHz. The operating frequency is selected depending on the free channels available to avoid interferences. The usage of this communication link allows sending data over TCP/IP up to 5 km line of sight. Therefore, the UAV can transmit live video streams and exchange information with the base station over a large range. Moreover, the communication range can be extended by configuring the devices to allow multi-hop. Therefore, the information can be sent to the other devices through other node when there is no direct connection.

As a central computer for the system we have selected a tablet running an Ubuntu distribution. A laptop with more processing capabilities could be used, but using a tablet offers the end user a very compact and easy platform to use the system. The end user can use this tablet to control the system, observe the collected data and to receive video streams. In addition, more computers, tablets or smartphones can connect to the system to receive the live data from the UAVs. In Figure 5-1 we depict a block diagram of the drones' hardware components.

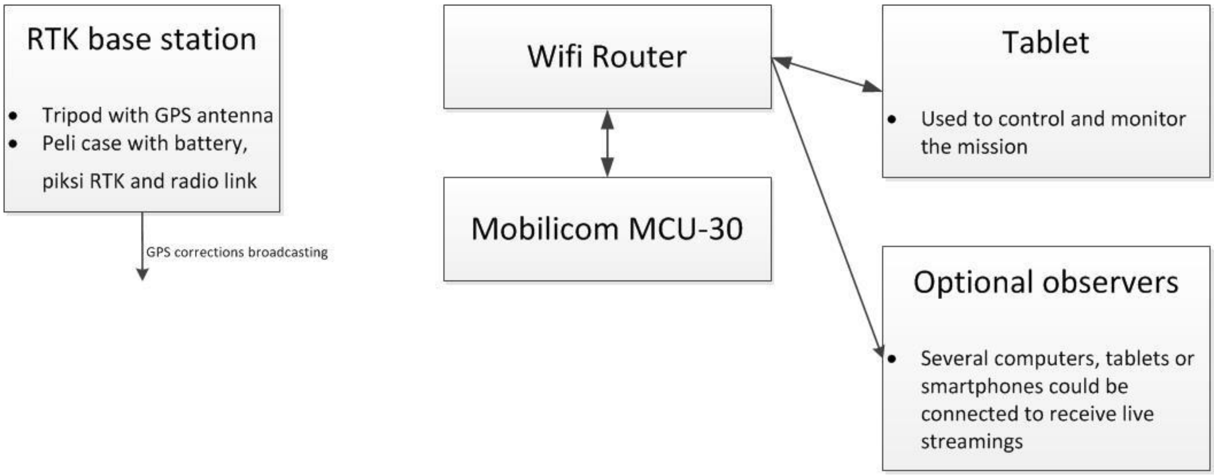


Figure 5-1: Block diagram describing the hardware included in the drones system base station.

#### 5.1.1.1.2 Software modules

Base station software allows an operator to command a swarm of MAVs to monitor an area of interest. Through a GUI that displays a map, an operator can define a region of interest (ROI) where a potential wildfire hotspot could be located. Once the operator performs this action, the system automatically generates routes that can be followed by the MAVs. The generation of routes follow two steps. First, points of interest (POI) are generated by the system. POIs are uniformly distributed over the ROI, and POIs separation is calculated according to the size of hotspot that an operator wants detect. Based on the POIs, routes are calculated. In particular, routes are calculated so that area coverage with time is maximized. . In the following sections, the main functionalities of the GUI will be presented, while a more comprehensive description of the GUI will be presented in deliverable D4.10. The essential advantage of this system is that it allows an operator to control multiple MAVs with a couple of clicks.

Once the base station is set up, the operator can use the tablet to monitor the system and drones' progress through a DronesGUI.This is possible as drones continuously send updates (drones status, pictures, video, and alarm signal in case a hotspot is detected). This update information is stored in a database. The database acts as a coordination mechanism for drones, as information from all drones is stored in the database. Additionally, the database acts as an interface to transmit required information to the HEIMDALL Service Platform (SP).

In Figure 5-2 we depict a block diagram of the base station software. In Section 5.1.2.1.2 we specify the modules' functionality, input, and output in detail.

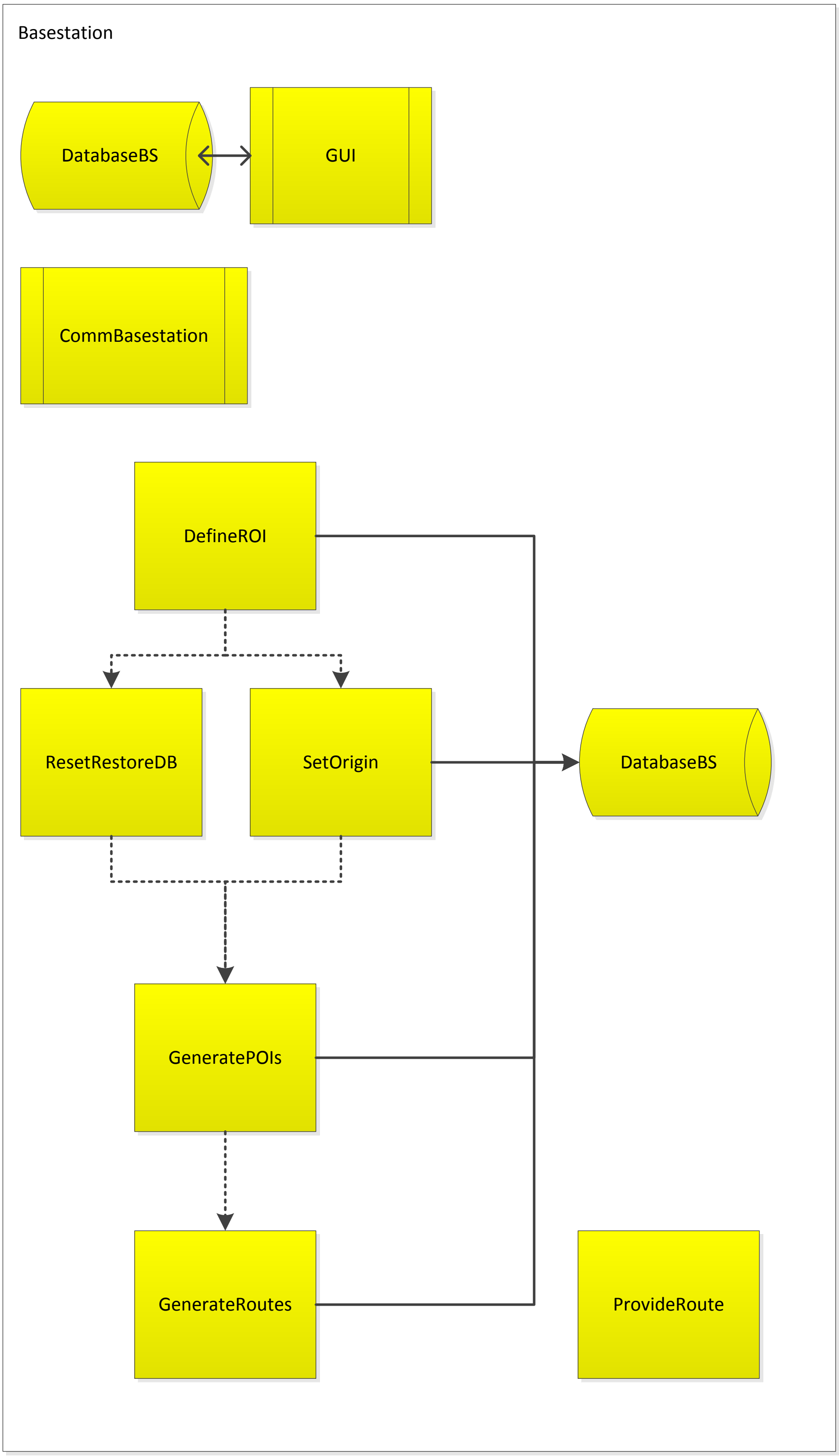


Figure 5-2: Swarm-Based Forest Fire Hotspots Detection. Base station software architecture.

#### 5.1.1.2 Drones

Drones are the second main component of our system. Here we understand drones as sensor carrying platforms. In the scope of this project sensors are equipped with the following principal sensors: RGB and thermal camera, with which the drones can identify potential hotspots.

To identify potential hotspots, drones must be able to fly to monitor the area of interest. Therefore, drones must be equipped with the hardware and software associated to this task. These are described in detail below.

#### *5.1.1.2.1 Hardware modules*

UAVs used in HEIMDALL are a custom design based on a DJI S900 frame [21]. Figure 5-3 shows the different hardware modules that are integrated in the UAV.

The flight controller of the UAV is a Pixhawk 2 [22]. This device controls the rotors of the UAV to stabilize it in the air and to move it to a desired coordinate. Pixhawk is an open hardware project that allows a lot of flexibility. We use Arducopter as the firmware, which is open source. This is an advantage over commercial flight controllers. Commercial flight controllers are typically closed and it is more difficult to integrate external hardware. The UAV positioning system is based on GNSS signals. It also incorporates a Piksi Multi RTK device to provide centimetre level accuracy when corrections from the base station are available.

The flight controller has limited computation capabilities. Therefore, we have integrated an Intel NUC [23] to be able to run our algorithms. The main task of this device is to handle and process video and images in order to detect hotspots. The Intel NUC also forwards the obtained video from the cameras to an HDMI video encoder, which provides highly compressed video while the delay on the image is kept under reasonable levels. The collected information and the status of the UAV are sent to the base station using the Mobilicom MCU-30 device that was explained in Section 5.1.1.1.1.

In order to connect the different systems we use a Raspberry Pi (RPi) [24]. This gives the system more flexibility. For example, when the computational capabilities of the RPi are enough for the application, the Intel NUC could be disassembled. The RPi is physically connected to the flight controller so they can exchange MAVlinks commands. The waypoint that is visited next is selected by the Intel NUC that sends this information to the RPi where it is translated to a MAVlink command that can be interpreted by the flight controller.

The Intel NUC and the HDMI video encoder are in a subnetwork and the RPi is the router to the main network. In that way, adding new network devices to the UAV does not affect the complete network.

The thermal images are collected using camera model Optris PI 400 [25]. It has a resolution of 382 x 288 pixels that allows us to see hotspot of 20 x 20 cm flying at 50 meters. It is connected by USB to the Intel NUC making it is possible to obtain full quality images and the temperature map of each pixel. The visual images are collected using a global shutter USB camera [26].

Both cameras are mounted on a servo gimbal to stabilize the image. It is possible to obtain better stabilization results by using brushless motor gimbals, but their weight and calibration effort is more in comparison to servo gimbals.

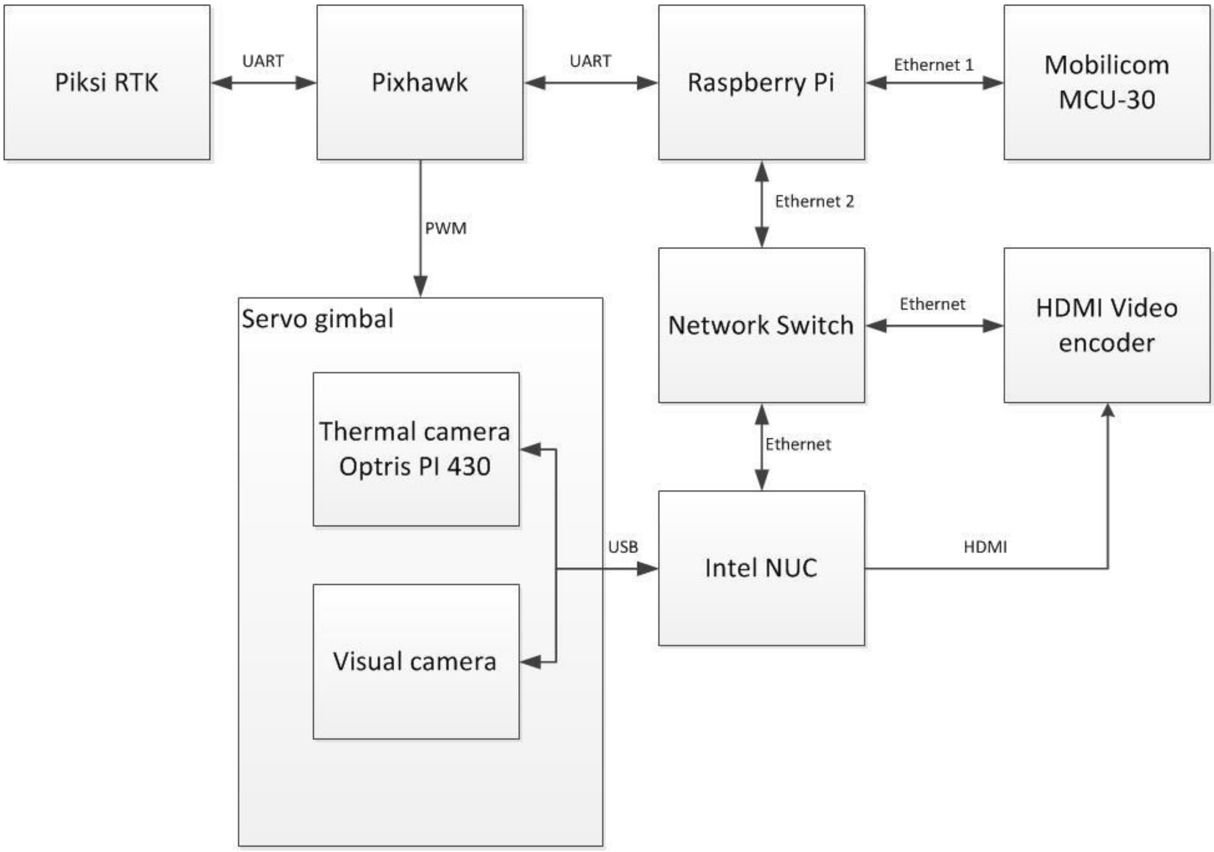


Figure 5-3 Block diagram of the hardware mounted on the UAV.

#### *5.1.1.2.2 Software modules*

Software modules specified in this section correspond to the software that runs on the drones. Once a drone is switched on, the software described here is automatically executed.

Figure 5-4, Figure 5-5, Figure 5-6 present block diagrams that correspond to the individual modules that constitute the drones' software t. The functionality, input and output of the modules is described in Section 5.1.2.2.2. The following paragraph describes an overall view of the drones' software functionality.

At the beginning of the mission, a drone has no information about the route it should follow to search for a potential hotspot. Therefore, the first step for a drone is to register with the base station, and then request a route that it can fly. The routes are assigned following the order on the corresponding DB table. In addition, collisions between routes are stores in the DB. Therefore, the drone is able to identify if the assigned route is colliding with a route that is already assigned and not finished. If that is the case, the drone will continue iterating through the table until it is able to get a non-colliding route.

Once a route is assigned to a drone, it will take off to fly to the first POI on the route and then to all the other POIs that belong to the group and will come back home and land once its mission is over. This process can continue in a loop. If, for example, we would like to continue flying with a drone, we only have to switch it off, change its battery, and then switch it on again.

As the drone flies the assigned route, it has two essential missions. Firstly, monitor the area to offer live information of the ROI to the operator. This live information consists of RGB and thermal pictures, and a live video stream. Secondly, the drone must identify potential hotspots. In the case a hotspot is identified, an alarm signal will be triggered.

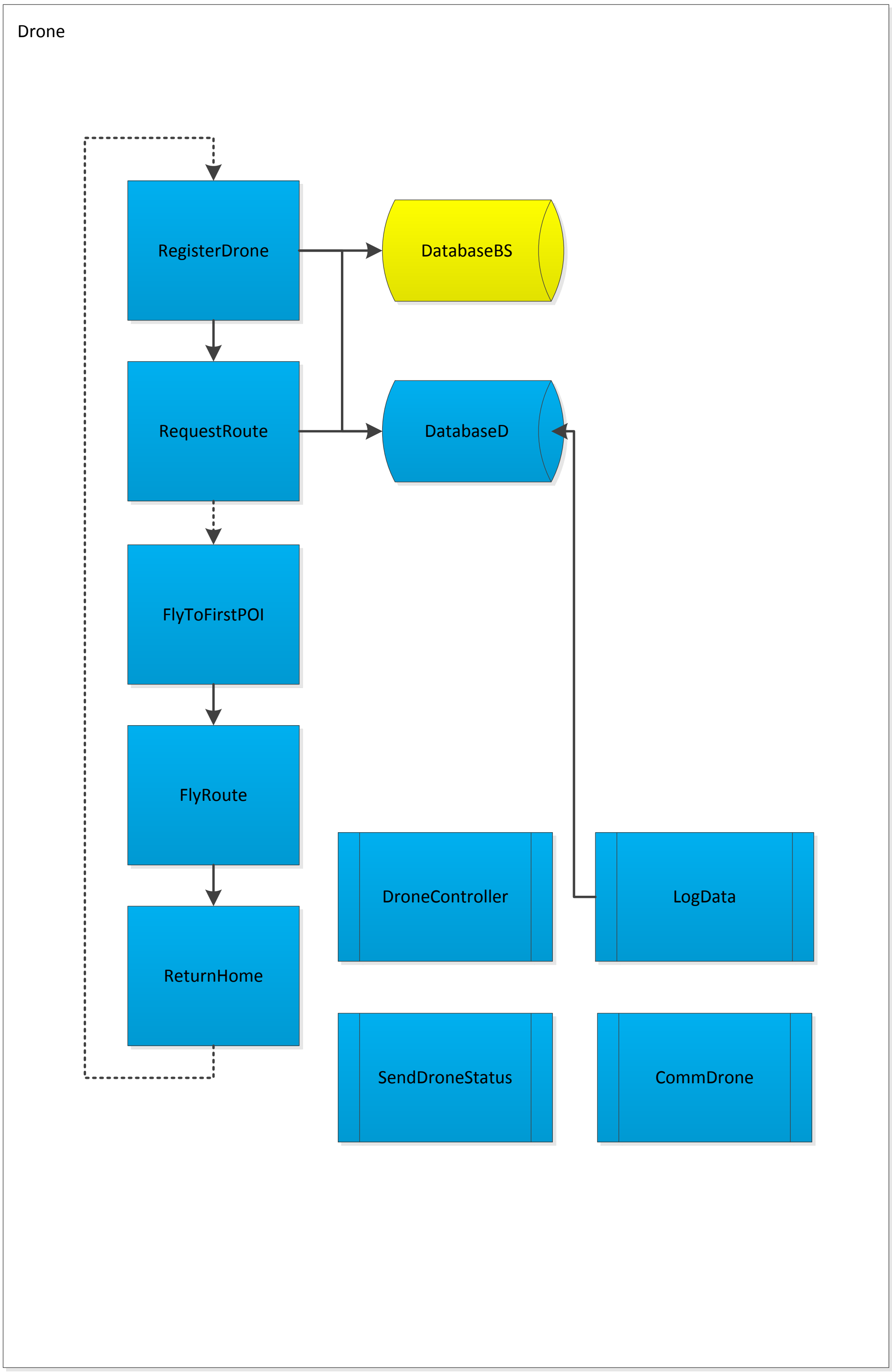


Figure 5-4: Swarm-Based Forest Fire Hotspots Detection. Drone software architecture.

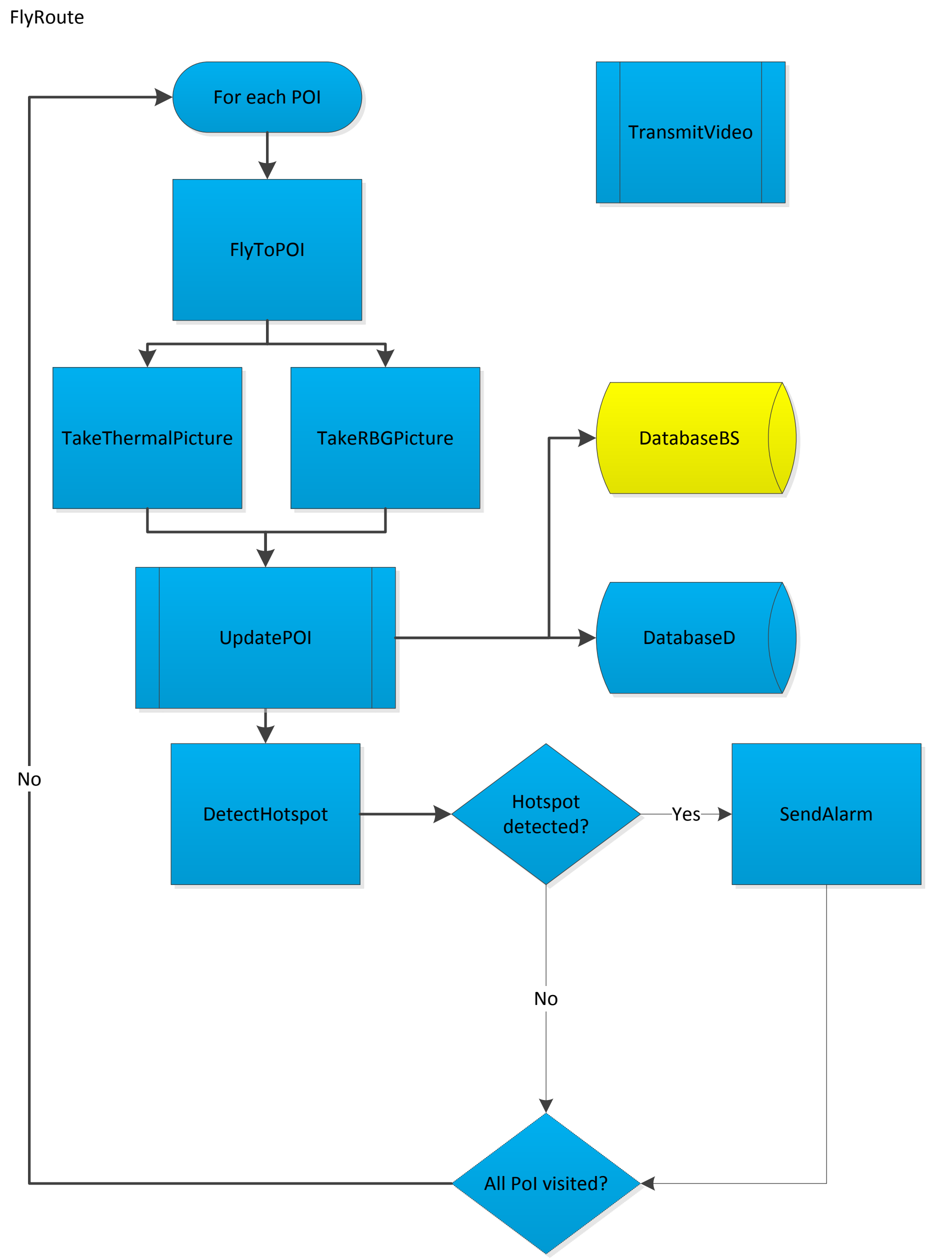


Figure 5-5: Swarm-Based Forest Fire Hotspots Detection. Base station software architecture FlyRoute.

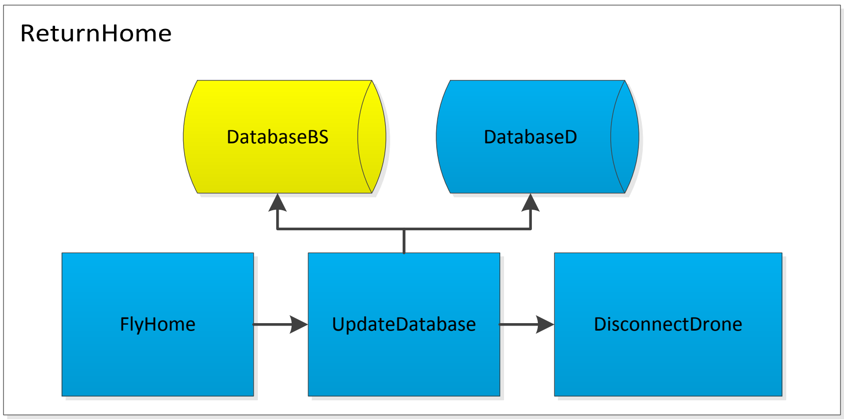


Figure 5-6: Swarm-Based Forest Fire Hotspots Detection. Base station software architecture ReturnHome.

## 5.1.2 Modules Specifications

In this section we specify in detail the hardware and software modules that constitute our drones system. For each of the individual modules we specify its functionality, input, and output below.

### 5.1.2.1 Base Station

#### *5.1.2.1.1 Hardware modules*

Base station hardware is composed of the following modules:

**a) RTK Base station**

- *Functionality*: improve positioning accuracy.
- *Input*: GPS coordinates of base station antenna.
- *Output*: GPS corrections.

**b) WiFi router**

- *Functionality*: creates a WiFi network for the base station devices (tablet, laptops).
- *Input and output*: network packages.

**c) Communication link**

- *Functionality (*Mobilicom MCU-30):
    - long range communication between UAV and base station (up to 5 km);
    - send/receive video streams; and
    - send/receive collected information.
- *Input and output*: network packages.

**d) Tablet**

- *Functionality*:
    - control the full system;
    - the user can introduce the mission needs;
    - store the collected information; and
    - display livestream video of the thermal and visual camera.
- *Input and output*: described in the software modules section.

**e) Optional observers**

- *Functionality*: devices such as smartphones, tablets or computers could connect to the network in order to receive the video stream from the UAVs. Those devices are not mandatory, and the system can work without them.
- *Input*: IP of the UAV that the user wants to monitor.
- *Output***:** live video stream.

#### *5.1.2.1.2 Software modules*

Base station software is composed by the following modules:

**a) CommBasestation**

- *Functionality*: communication software and hardware needed to communicate with CommDrone.
- *Input*: TCP/IP packages sent by the drones.
- *Output*: TCP/IP packages sent to the drones.

**b) DatabaseBS**

- *Functionality*:
  - Store the following information:
    - drones: position, time, state, IP, ID;
    - POIs: position, time, state (visited, unvisited), drone visitor, thermal pictures (time and geo-referenced), RGB pictures (time and geo-referenced);
    - routes: reference to a set of POIs, state (assigned, unassigned, completed, uncompleted), drone in charge;
    - hotspots: status (to be monitored, outdated), reference to POI, temperature vector;
    - home area;
    - system parameters;
    - ROI: ordered set of vertices of a polygon; and
    - log data.
  - API that :
    - allows drones to store information;
    - allows drones to retrieve information;
    - manages writing/reading conflicts;
    - allows us to backup/restore the DB; and
    - generates an DB with empty tables.
- *Input*:
  - information defined in functionality (to be stored); and
  - command to retrieve information (e.g. last 10 POIs visited by drone B).
- *Output*:
  - confirmation that information was correctly stored/retrieved.

**c) DefineROI**

- *Functionality*:
  - take user input and store it in DB;
  - set flying height base on a minimum, maximum and a measure of the time to cover the area for that given height, and minimum hotspot that can be detected.
- *Input*:
  - polygon in a map, point by point;
  - minimum flying height; and
  - maximum flying height.

- *Output*:
    - confirmation message.

**d) GeneratePOIs**

- *Functionality*:
    - generate POIs in the region of interest;
    - define a grid over the ROI;
    - calculate the grid's cell size (given the flying height and camera parameters) so that POIs guarantee a full coverage of the ROI; and
    - initialize POIs as unvisited.
- *Input*:
    - ROI;
    - flying height; and
    - camera parameters.
- *Output*: POIs.

**e) GenerateRoutes**

- *Functionality*: generate routes to allow UAVs to visit POIs. Routes shall be optimal in terms of speed to visit all POIs with minimum cross among routes. A vehicle routing problem formulation is our proposed and implemented method to solve the problem [4]. In addition, it is possible to calculate the route using a meander strategy.
- *Input*:
    - POIs;
    - criteria to optimize; and
    - maximum flying time of drones.
- *Output*: matrix specifying calculated routes.

**f) GUI**

- *Functionality*:
    - start/stop the whole system;
    - provide telemetry information and real time video of selected UAV;
    - display visual and thermal picture of hotspot (provided by DB);
    - take user input regarding if a hotspot should be monitored or not and write it into DB (hotspot is monitored by default);
    - provide information about the progress of exploration;
    - display UAVs' position;
    - allow user to introduce a ROI;
    - allow user to introduce origin (where UAVs will land and take off);
    - provide an interface to the DB, in order to store ROI and home;
    - display RGB and thermal pictures of visited POIs; and
    - provide a return home button that triggers ReturnHome.
- *Input*:
    - user inputs;
    - UAVs inputs; and
    - DB data from every table.
- *Output*: displayed information as described in functionality.

**g) RegisterDrone**

- *Functionality*: register drone in DB, and store drone's parameters.
- *Input*: drone information (position, time, state, IP, ID).
- *Output*: confirmation message.

**h) ResetRestoreDB**

- *Functionality*: according to user preference, this module shall initialize a DB with empty tables, or restore a previous DB.
- *Input*: user decision (reset/restore).
- *Output*: confirmation that DB was reset/restored.

**i) SetOrigin**

- *Functionality*: set position around which UAVs will be placed for start and take off.
- *Input*: user click on a map.
- *Output*: GPS coordinat

**j) UpdateHeimdallSP**

- **Functionality: It reads from the base station DB using DatabaseBS and convert it to json files according to the SP specifications. These json files are sent to the SP using *requests* python package. It runs completely independent from the rest of the software packages and it can run in a different device, e.g., a dedicated laptop. Using a dedicated laptop simplifies the network configuration.**
- **Input: DB tables containing information about drones positions, visual and thermal images and detected hotspots.**
- **Output: json files containing information about drones positions, visual and thermal images and detected hotspots.**

### 5.1.2.2 Drones

#### *5.1.2.2.1 Hardware modules*

Hardware of the drones is composed of the following modules:

**a) Pixhawk**

- *Functionality*: flight controller of the UAV:
  - Stabilize the UAV in the air and
  - Move it to the desired locations
- *Input*:
  - GPS signal;
  - RTK position;
  - commands from the Raspberry PI; and
  - commands from the remote controller.
- *Output*:
  - MAVlink messages to the Raspberry Pi; and
  - PWM signal to control the servos.

**b) Piksi RTK**

- *Functionality*: read GPS corrections that are sent by the RTK base station to provide a centimetre level accuracy positioning.
- *Input*: GPS corrections.
- *Output*: coordinates of the UAV.

**c) Raspberry PI**

- *Functionality*:
  - translate MAVlinks commands coming from the Pixhawk into ROS topics and vice versa;

    - router to forward packages between the main network and the subnetwork of the payloads.
- *Input*:
    - MAVlink messages from Pixhawk; and
    - network packages.
- *Output*:
    - MAVlink commands; and
    - network packages.

**d) Intel NUC**

- *Functionality*:
    - process thermal and visual images; and
    - run the robot operating system (ROS) master.
- *Input*:
    - ROS topics from the Raspberry Pi;
    - video from thermal camera; and
    - video from thermal camera.
- *Output*:
    - HDMI video with either thermal o visual camera;
    - ROS messages with commands for the UAV; and
    - images from thermal and visual camera.

**e) HDMI Video encoder**

- *Functionality*: produce high quality video streams using low bandwidth based on the HDMI signal that it receives.
- *Input*: HDMI video from Intel NUC with either thermal or visual images.
- *Output*: live video stream in standard format (Real Time Streaming Protocol is currently used).

**f) Mobilicom MCU-30**

- *Functionality*: extend the communication range up to 5 km using TCP/IP.
- *Input*:
    - network packages; and
    - video stream.
- *Output*:
    - network packages; and
    - video stream.

**g) Thermal camera Optris Pi 430**

- *Functionality*: provide thermal images via USB connection.
- *Input*: configuration parameters.
- *Output*: thermal images.

**h) Visual camera**

- *Functionality*: provide visual images via USB connection.
- *Input*: configuration parameters.
- *Output*: visual images.

**i) Servo gimbal**

- *Functionality*: stabilize the cameras while the UAV moves.
- *Input*: PWM signal.
- *Output*: -

#### 5.1.2.2.2 Software modules

Software of the drones is composed of the following modules:

**a) CommDrone**

- *Functionality*: provide drone with TCP/IP communication capabilities, both inter-drone and with base station.
- *Input*: -
- *Output*: -

**b) DatabaseD**

- *Functionality*: same functionality as DatabaseBS, except ROI that does not need to be known by the drone.
- *Input*: same as DatabaseBS.
- *Output*: same as DatabaseBS.

**c) DetectHotspot**

- *Functionality*: process thermal and RGB images to obtain hotspot location, temperature, and size. Identification of a hotspot is done by thresholding.
- *Input*: thermal and RGB images.
- *Output*: hotspot location, temperature, and size.

**d) DisconnectDrone**

- *Functionality*: set drone as inactive, and back up DatabaseD.
- *Input*: drone to be disconnected.
- *Output*: confirmation message.

**e) DroneController**

- *Functionality*: guide drone towards commanded waypoint.
- *Input*: waypoint / set of waypoints, and drone's telemetry.
- *Output*: confirmation message of waypoint / mission accomplished.

**f) FlyHome**

- *Functionality*: send waypoints to drone to reach home position while avoiding collisions with other drones. This functionality is identical to FlyToFirstPOI.
- *Input*: same as in FlyToFirstPOI.
- *Output*: same as in FlyToFirstPOI.

**g) FlyRoute**

- *Functionality*: it encompass following modules:
  - FlyToFirstPOI
  - **FlyToPOI**
  - take pictures (RGB and thermal) and transmit them (TakeRGBPicture and TakeThermalPicture);
  - transmit video (TransmitVideo);
- detect hotspot (DetectHotspot);*Input*: specified in the individual submodules in section 5.1.2.2.2.
- *Output*: specified in the individual submodules in section 5.1.2.2.2.

**h) FlyToFirstPOI**

- *Functionality*: permit a drone to fly to the first point in a route while avoiding collisions with other drones.
- *Input*: allowed flying areas from drones starting position and first POI.
- *Output*: confirmation message.

**i) FlyToPOI**

- *Functionality*:
  - fly from A (current position) to B (goal POI) in a straight line using the drone controller; and
  - check that *position B* was reached with a certain accuracy, and stop drone.
- *Input*:
  - drone current position;
  - goal POI; and
  - accuracy to reach goal POI.
- *Output*: confirmation message that *position B* was reached.

**j) LogData**

- *Functionality*: log system information for debugging, and data post-processing.
- *Input*: information to be logged.
- *Output*: data files.

**k) ProvideRoute**

- *Functionality*: provide drone a non-inspected route.
- *Input*: request for a route.
- *Output*: available route that does not incur in inter-drones collisions.

**l) RequestRoute**

- *Functionality*: request a non-visited route, which does not incur in collisions with other drones.
- *Input*: drone ID.
- *Output*: confirmation message.

**m) ReturnHome**

- *Functionality*: trigger FlyHome, UpdateDatabase, and DisconnectDrone.
- *Input*: -
- *Output*: -

**n) SendAlarm**

- *Functionality*:
  - send an alarm message to base indicating that a hotspot was detected; and
  - send information related to hotspot (location, temperature, size, pictures) to base.
- *Input*: hotspot information (location, temperature, size, pictures).
- *Output*: confirmation message from base station indicating that alarm was received.

**o) SendDroneStatus**

- *Functionality*: send to base station drones information (position, ID, and time stamp).
- *Input*: drone's position, ID and time stamp.
- *Output*: -

**p) SendPicturesToBase**

- *Functionality*:
  - transmit DB table that includes POIs together with the respective pictures, and POIs info; and

    - retrigger itself periodically to transmit those pictures that could not be transmitted due to failures in communication link.
- *Input*:
    - DB tables with POIs information and pictures;
    - retransmission interval;
    - maximum number of retransmissions; and
    - retransmission queue length.
- *Output*: DB table.

**q) TakeRGBPicture**

- *Functionality*: take a geo-referenced and time-stamped RGB picture.
- *Input*:
    - camera parameters (resolution, brightness…);
    - drone's state; and
    - time stamp.
- *Output*: geo-referenced time-stamped RGB image.

**r) TakeThermalPicture**

- *Functionality*: take a geo-referenced and time-stamped thermal picture.
- *Input*:
    - camera parameters (resolution, brightness…);
    - drone's state; and
    - time stamp.
- *Output*: geo-referenced time-stamped thermal image.

**s) TransmitVideo**

- *Functionality*: transmit RGB or thermal video from a drone as specified by user.
- *Input:*
    - type of video to be transmitted (RGB or thermal);
    - drone of interest; and
    - video quality, resolution, and format.
- *Output*: video.

**t) UpdateDatabase**

- *Functionality*: update base station DB with drones' information that could not be transmitted online.
- *Input*: drones' DB.
- *Output*: confirmation message.

## 5.2 GB-SAR-Based Landslide Monitoring

In the following, the different elements of the GB-SAR landslide monitoring are specified.

### 5.2.1 System Overview

The GB-SAR Landslide Monitoring system is designed to estimate the displacement of the selected slope where the landslide occurs. This data is provided to the users as close to real time as possible, especially when it is used in addition to a warning and alarm system.

The system is a commercial instrument available from IDS spa, named Ibis-L [2]. A SW is available from the supplier to start the acquisition with a specific GUI implemented using MATLAB®, while data are processed by using a home-made processing chain developed in C, C++, IDL® and MATLAB® by CTTC. The DATABASE can be analysed by ENVI® and shown by QGIS®. The apparatus is composed of the following modules (see Figure 4-1):

#### 5.2.1.1 Radar System

The radar system, is the core of the measuring system. It is configured at the beginning of the survey with parameters set, which stands for all the acquisition, according to users' requirements. The parameters to be set are:

- Range resolution (meters);
- Azimuth resolution (radians);
- Image size (meters x meters), maximum range
- Acquisition period; depending on deformation characteristics.

These parameters determine the features of the products. The acquisition is fully automatic and provides in standard conditions data update every 10 minutes (continuous case). The acquisition configuration can be changed on user demand, and a history of the different configurations is stored in the local PC.

#### 5.2.1.2 Power unit

The power unit is composed of an apparatus able to provide the power for the radar transceiver, the rail (mechanical motion) and link devices.

#### 5.2.1.3 Processing unit

Data from the sensor (automatic measurements) are performed with a data acquisition internal to the apparatus and fully scheduled at the beginning of the survey. An internal SW controls the execution of the procedures for the measurements, storage, and transmission of the data. The raw data are processed by internal software not accessible to the user.

#### 5.2.1.4 Communication interface (local and receiving portal)

Collected data are sent to the server via a communication interface, usually the gateway of a wireless link. The communication protocols supported by the interfaces are diverse: USB interface or wireless 3G/4G or satellite communications in remote areas. All the systems in the context of this project work in a push way, so data is automatically pushed by the base station to the server located at CTTC.

#### 5.2.1.5 Database

Received radar data, (matrices of complex number and maps, acquired through the GB-SAR system reaches the server, where they are stored in a database that is updated every new acquisition and processing. Processed data is available at CTTC server and to the user through a FTP to the Service Platform in .csv files (temporal series) or .jpg files (quick look) or shape files. The user can access the process data in the server through a login. The processed data include deformation maps, and temporal series of deformations obtained after processing the radar images acquired during the concerned period.

### 5.2.2 Modules Specifications

a) **The radar system**
   - *Functionality*: measure the radar reflectivity.

- *Input*: configuration parameters.
- *Output*: raw data (radar received signal to be processed).

**b) The power unit**

- *Functionality*: provide different voltages to the system.
- *Input*: start of acquisition.
- *Output*: state of the battery/power line.

**c) The processing unit**

- *Functionality*: control the acquisition and process radar data using operator expertise and output quality parameters as for instance temporal coherence of the images set.
- *Input*: configuration parameters. Processing software.
- *Output*: radar maps.

**d) Local portal**

- *Functionality*: establish a link to transfer data from local PC to CTTC server.
- *Input*: address configuration (IP or another identifier depending on the selected link). Start command from user for sending data.
- *Output*: connection available between measuring system and CTTC server.

**e) Receiving portal**

- *Functionality*: receive data from local PC.
- *Input*: address configuration (IP or another identifier depending on the selected link). Command from user to accept link and data transfer.
- *Output*: connection available between measuring system and CTTC server. Data available at CTTC server

**f) The database (accessible to the users)**

- *Functionality*:.

  The data hosted by CTTC is stored in a server publicly available running an on-premises file sharing and collaboration platform, namely "NextCloud" (see https://nextcloud.com/). There, a shared folder was set up to store data in the aforesaid server. To connect, retrieve or store new data from / to this shared folder, the WebDaV protocol over https must be used - this is a standard service offered by NextCloud.

  The URL to access the service is https://cloud.cttc.es/remote.php/webdav/. The service is private, accessible only by authorized personnel working for the HEIMDALL project. Thus, credentials have been provided to those having to manage CTTC's HEIMDALL data.

  Both the Windows and Linux operating systems are capable of mounting a WebDav shared folder as a regular network drive, thus making possible to manage the information stored in the CTTC server as if it were in a local disk attached to the computer willing to manage such data. This is, in fact, the recommended procedure. The steps to mount a WebDav share are widely available for both the Windows and Linux operating systems, so accessing CTTC's data should be a rather easy issue to accomplish.

- *Input*: data received from measuring system. Database rationale.
- *Output*: final product available to the user.

## 5.3 Geotechnical/Hydrological Landslide Monitoring

### 5.3.1 System Overview

The geotechnical and hydrological landslide monitoring systems are designed to measure physical properties related to the stability of the terrain in areas that have been identified as notably susceptible to the occurrence of landslides. These data are useful to warn about approaching unstable conditions that can trigger landslides.

The system consists of four main components: sensors, loggers, a communication interface and the internal server. The data is gathered by the sensors and directly stored in the loggers, where the sensors are connected by wires. The loggers transmit the data (wired or wirelessly) to the server through a communication interface (Figure 5-7).

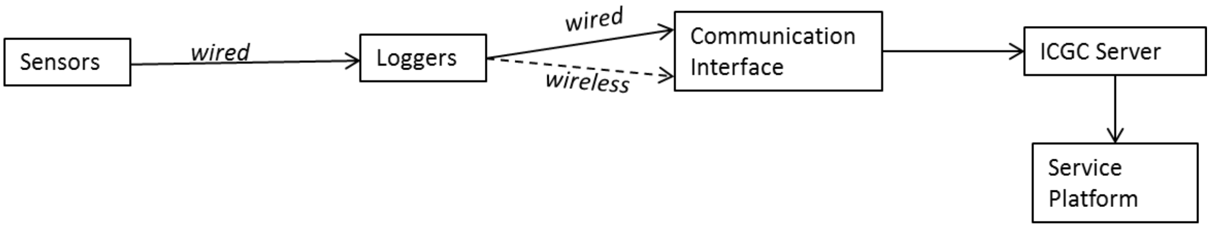


Figure 5-7: System overview containing all the components and the communication between them

### 5.3.2 Modules Specifications

#### 5.3.2.1 Sensors and measuring devices

Sensors and measuring devices should be selected based on the type of terrain movement, its triggering factors, its velocity, etc. However, the most decisive factor is often the budget available. Sensors and measuring devices are intended to monitor factors related to the stability conditions of the terrain.

There are many different classification systems for sensors and measuring devices, based on different parameters [18]. From a technical point of view, a usual categorization is based on the type of output they provide. According to this, sensors and measuring devices can be:

- Manual measurements
- Automatic measurements
  - Analogue output
  - Digital output

At the sites checked for the development of this project [26], most common manual measurements are performed using:

- Tell-tales to monitor crack aperture. Tell-tales consist of two plates which overlap for part of their length. One plate is calibrated in millimetres and the overlapping plate is transparent and marked with a hairline cursor. As the crack width opens or closes, one plate moves relative to the other [27]
- Divers to monitor water level. Divers are sensors to measure and register the groundwater level and ground water temperature [28]

And the most common automatic measurements are done using the following sensors:

- Analogue:

    - Crackmeters to measure the aperture of cracks. Crack meters are intended to measure the movement across surface cracks and joints. They are installed by anchoring two threaded anchors with ball joints on opposite sides of the crack [29].
    - Piezometers to measure groundwater pore water pressure [30].
- Digital:
    - Tiltmeters to measure the tilt of buildings or structures [32].

For the HEIMDALL platform, as there were no in-situ sensors available for the case study (i.e. Monesi), the network already implemented by ICGC in Barberà de la Conca was used as an example of how these devices can help on a landslide emergency. In this village a 175 m long crack has been opening since 2010, crossing the whole village. This network is composed by 21 crackmeters, 3 cable extensometers, 2 piezometers, 2 tiltmeter and their corresponding power supplies (Table 5-1). There is also an Automatic Weather Station (AWS) which records temperature and rainfall.

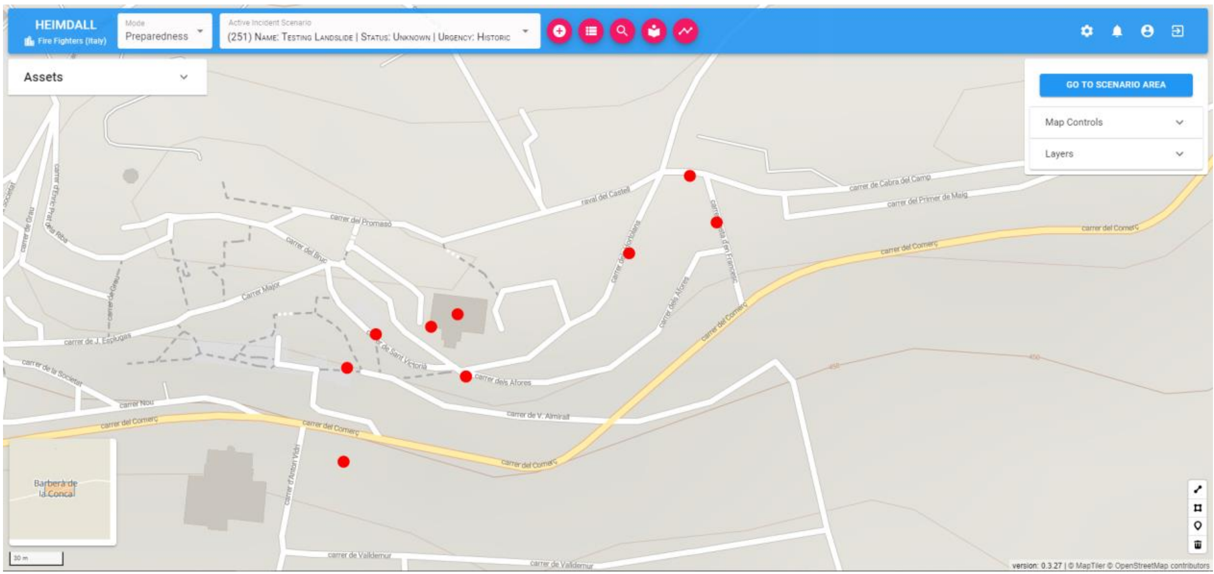


Figure 5-8: Barberà de la Conca In-Situ sensors network.

Table 5-1: Barberà de la Conca's In-Situ sensors network specifications.

| Type of sensor | Sensor code | Friendly sensor's name | Manufacturer | Model |
|---|---|---|---|---|
| Crackmeter | LE_11_FO | Crackmeter - orthogonal 11 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | LE_12_FL | Crackmeter - longitudinal 12 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | LE_13_FL | Crackmeter - longitudinal 13 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | ES21_FO | Crackmeter - longitudinal 21 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES22_FL | Crackmeter - longitudinal 22 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES23_FV | Crackmeter - orthogonal 23 | Sisgeo | 0D313SA5000 |
| Crackmeter | LE_31_FO | Crackmeter - orthogonal 31 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | LE_32_FL | Crackmeter - longitudinal 32 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | LE_33_FL | Crackmeter - longitudinal 33 | novotechnik | TX2-0100-716-002-202 |

| | | | | |
|---|---|---|---|---|
| Crackmeter | ES41_FO | Crackmeter - longitudinal 41 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES42_FL | Crackmeter - longitudinal 42 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES43_FV | Crackmeter - longitudinal 43 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES51_FO | Crackmeter - orthogonal 51 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES52_FL | Crackmeter - longitudinal 51 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES53_FV | Crackmeter - vertical 53 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES61_FO | Crackmeter - orthogonal 61 | Sisgeo | 0D313SA5000 |
| Crackmeter | PO_61_FO | Crackmeter - orthogonal 61 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | PO_62_FL | Crackmeter - longitudinal 62 | novotechnik | TX2-0100-716-002-202 |
| Crackmeter | ES81_FO | Crackmeter - orthogonal 81 | Sisgeo | 0D313SA5000 |
| Crackmeter | ES82_FL | Crackmeter - orthogonal 82 | Sisgeo | 0D313SA5000 |
| Crackmeter | PO_81_FO | Crackmeter - orthogonal 81 | novotechnik | TX2-0100-716-002-202 |
| Cable Extensometer | ES11_CO | Extensometer- orthogonal 11 | Glötzl | GDEX |
| Cable Extensometer | LE_21_CO | Extensometer - orthogonal 21 | Sisgeo | 0D313FA5000 |
| Cable Extensometer | ES31_CO | Extensometer - orthogonal 31 | Sisgeo | 0D313FA5000 |
| Piezometer | S22_1 | Piezometer pressure | Impress (Mapro) | D-IMCL |
| Piezometer | S22_2 | Piezometer pressure | Impress (Mapro) | D-IMCL |
| Tiltmeter | ES_71_CN | North tilt | Sisgeo | 0S542MA1000 |
| Tiltmeter | ES_72_CW | West tilt | Sisgeo | 0S542MA1000 |
| Thermometer | PTemp | Air Temperature | Campbell | T107 |
| Rain gauge | Pluja_Tot | Rainfall | Young | 52202 |

#### 5.3.2.2 Loggers

Measurements from sensors (automatic measurements) are performed using loggers. These devices normally power the sensor, read and store the output data, and depending on the type of monitoring system, send the data to a base station.

In case of the wireless monitoring systems, the loggers capture data from the sensors and send it to the gateway (base station). The gateway acts as communication interface and sends data to the server [33].

In case of wired monitoring systems, the logger itself is the base station, and the sensors are directly connected to it. The logger is, in this case, the communication interface [31].

In Barberà de la Conca network there are three dataloggers installed manufactured by Campbell, two cr800 and one cr1000.

#### 5.3.2.3 Communication interface

Data collected by loggers is sent to the server via a communication interface. Sometimes it is the gateway of the wireless monitoring system, and, in case of wired monitoring systems, it is the logger itself.

The communication protocols supported by the interfaces are diverse: from 3G/4G, Wi-Fi, voice calls or satellite communications in remote areas. In the case of the study sites for this project only 3G/4G communication is used through a Sierra LS300 router. Data is automatically pushed by the base station to the server.

#### 5.3.2.4 ICGC Server

Data from the sensors and measuring devices reach the server in different ways, but it is stored in a database, which is updated every new acquisition and processing. Raw data, gathered from sensors, is stored in the server. Raw data from vibrating wire sensors (crackmeters or piezometers) is gathered in ohms, and it is automatically transformed into Hz. Since this data has no physical sense, it is also transformed into data in engineering units (mm for crackmeters and kPa for piezometers). Also, these data are sent and stored in the server.

Data is available outside of the server via a REST API that periodically obtains this data. The data can be accessed through a GET method request indicating the station id and the time period requested (start and end time):

```
GET /services/rest/sensorsapi/sensors/<station id> from= <initial date.time> & to= <end date.time> HTTP/1.1
```

Once the information is got from the server it is sent back to the SP through a POST method:

```
POST: {"id":<station_id>,"name":<station_name>,"description":<friendly station name> ,"history": [{"timestamp":"2019-03-11T12:00:00",  "value": "0.031 graus"},
…
,{"timestamp":"2019-03-15T13:00:00","value":"0.03 graus"}]}
```

As an example, below, the results got from the northern tiltmeter is shown (Figure 5-9).


```
{"id":34,"name":"ES_71_CN","description":"North tilt","history":[{"timestamp":"2019-03-11T12:00:00","value":"0.031 graus"},{"timestamp":"2019-03-11T12:30:00","value":"0.036 graus"},{"timestamp":"2019-03-11T13:00:00","value":"0.03 graus"},{"timestamp":"2019-03-11T13:30:00","value":"0.025 graus"},{"timestamp":"2019-03-11T14:00:00","value":"0.03 graus"},{"timestamp":"2019-03-11T14:30:00","value":"0.023 graus"},{"timestamp":"2019-03-11T15:00:00","value":"0.029 graus"},
…
,{"timestamp":"2019-03-15T11:30:00","value":"0.036 graus"},{"timestamp":"2019-03-15T12:00:00","value":"0.034
```

```
graus"},{"timestamp":"2019-03-15T12:30:00","value":"0.031
graus"},{"timestamp":"2019-03-15T13:00:00","value":"0.03 graus"}]}
```


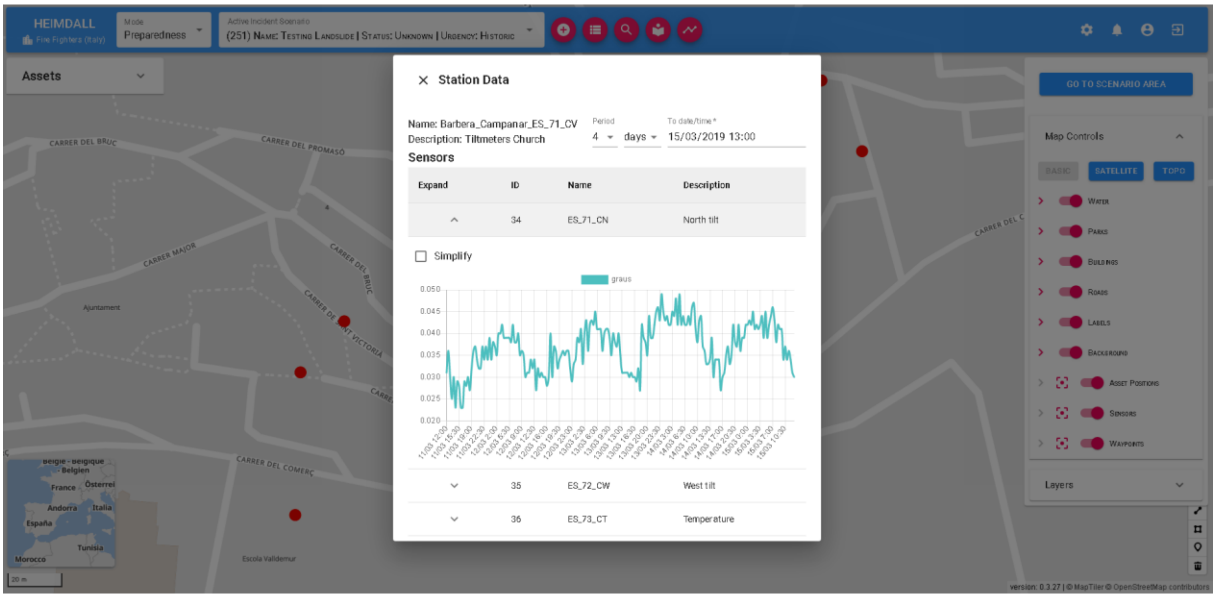


Figure 5-9: Barberà de la Conca In-Situ sensors network.

# 6 Test Plan

In this section we introduce a test plan for each of the considered sensors. The presented tests were designed to meet with technical requirements listed in Section 2.

## 6.1 Swarm-Based Forest Fire Hotspots Detection

The objective of this test plan is to validate the integration of the drone sensors module into the HEIMDALL platform.

Table 6-1: Drones test plan. TS_DataSituDrones_1.

| | |
|---|---|
| **Test ID** | *TS_ DataSituDrones_1* |
| **Requirements to be verified** | • *TR_DataSitu_6*<br>• *TR_DataSitu_7* |
| **Test objective** | Verify that an operator can select a ROI, for which POIs will be automatically generated by the system. |
| **Test procedure** | 1. The user starts the GUI, where a map centred at a simulated location is displayed.<br>2. The user clicks on the map, and the ROI is created.<br>3. The user presses the POI generation button, and POIs are automatically created. |
| **Test prerequisites/ configuration** | • The GUI is running.<br>• The operator location is valid; i.e. it is a real location.<br>• The ROI is defined by a minimum of three vertices. |
| **Success criteria** | • POIs are displayed in the GUI.<br>• POIs are initialized as unvisited. |
| **Results analysis** | Multiple flying tests with up to 4 drones have been carried out to verify the correct functioning of the system. |
| **Success** | PASSED /Tests were successfully accomplished for release C |

Table 6-2: Drones test plan. TS_DataSituDrones_2.

| | |
|---|---|
| **Test ID** | *TS_ DataSituDrones_2* |
| **Requirements to be verified** | • *TR_DataSitu_10* |
| **Test objective** | Verify that drones are able to send information to base station, and store this information in base station DB. |
| **Test procedure** | 1. Drones are sent to some pre-defined location.<br>2. The user connects to drones to trigger cameras to take pictures.<br>3. The user connects to drones to send pictures. |
| **Test prerequisites/ configuration** | • Drones are set up to fly and take pictures.<br>• Drones are correctly configured to permit a remote log in.<br>• Base station GUI is running. |
| **Success criteria** | • Pictures are displayed in base station GUI. |

| | |
|---|---|
| **Results analysis** | Multiple flying tests with up to 4 drones have been carried out to verify the correct functioning of the system. |
| **Success** | PASSED /Tests were successfully accomplished for release C |

Table 6-3: Drones test plan. TS_DataSituDrones_3.

| | |
|---|---|
| **Test ID** | *TS_ DataSituDrones_3* |
| **Requirements to be verified** | • *TR_DataSitu_9* |
| **Test objective** | Verify that a drone is able to autonomously identify a hotspot by analysing RGB and thermal pictures. |
| **Test procedure** | 1. Generate artificial hotspots, by burning some materials.<br>2. Send drone to a specified location where there is or there is not a hotspot.<br>3. The user connects to drones to trigger cameras to take pictures.<br>4. The user connects to run the hotspot detection algorithm. |
| **Test prerequisites/ configuration** | • Drones are set up to fly and take pictures.<br>• Drones are correctly configured to permit a remote log in.<br>• Drone hotspot detection algorithm is running. |
| **Success criteria** | • A hotspot is identified if there is one.<br>• A hotspot is not identified if there are no hot spots. |
| **Results analysis** | Multiple flying tests have been carried out to verify the correct functioning of the system. A dedicated measurement campaign together with colleagues from INT was organized in Girona (Spain) to verify this feature. Results of this campaign were published in [34]. |
| **Success** | PASSED /Tests were successfully accomplished for release C |

Table 6-4: Drones test plan. TS_DataSituDrones_4.

| | |
|---|---|
| **Test ID** | *TS_ DataSituDrones_4* |
| **Requirements to be verified** | • *TR_DataSitu_11* |
| **Test objective** | Verify that a drone is able to transmit a video. |
| **Test procedure** | 1. Command drone to fly a specified route.<br>2. The user connects to drones to trigger video transmission.<br>3. The user starts base station GUI. |
| **Test prerequisites/ configuration** | • Drones are set up to fly and take pictures.<br>• Drones are correctly configured to permit a remote log in.<br>• Base station GUI is running. |
| **Success criteria** | • Video is displayed in the DronesGUI.<br>• User can select drone, and RGB or thermal video. |
| **Results analysis** | Multiple flying tests with up to 4 UAVs have been carried out to verify the correct functioning of the system. |
| **Success** | PASSED /Tests were successfully accomplished for release C |

## 6.2 GB-SAR-Based Landslide Monitoring

The objective of this test plan is to validate the integration of the in-situ sensors module into the HEIMDALL platform.

Table 6-5: GB-SAR test plan. TS_DataSituMon_1.

| | |
|---|---|
| **Test ID** | *TS_ DataSituMon_1* |
| **Requirements to be verified** | ○ *TR_DatSituMon_1*<br>○ *TR_DatSituMon_2* |
| **Test objective** | Verify that the SP provides a file where GB-SAR Images are able to show the difference between stable and instable areas.<br>Verify that the SP provides the files acquired with the selected acquisition interval. |
| **Test procedure** | 1. The user connects to the HEIMDALL VPN.<br>2. The user starts the web portal and logs in.<br>3. The user opens the main page and visualise the images set available for the selected site.<br>4. The user clicks on one of the two options windows to display the deformation map or the temporal series. |
| **Test prerequisites/ configuration** | • The web portal needs to be up and running.<br>• The service platform can serve the requested data. |
| **Success criteria** | The user can visualise the following elements on the SP GeoServer:<br>• Original data (Maps /Temporal series).<br>• New data. |
| **Results analysis** | The test has been performed and passed visualising maps, and plots already stored was tested in Release A. |
| **Success** | PASSED |

## 6.3 Geotechnical/Hydrological Landslide Monitoring

The objective of this test plan is to validate the integration of the in-situ sensors module into the HEIMDALL platform.

Table 6-6: Geotechnical/hydrological monitor test plan. TS_DataSituGeo_1.

| | |
|---|---|
| **Test ID** | *TS_ DataSituGeo_1* |
| **Requirements to be verified** | • *TR_DataSituGeo_1*<br>○ *Sys_DataSitu_1*<br>• *TR_DataSituGeo_3*<br>*Sys_DataSitu_1* |
| **Test objective** | The user can visualize the historical and new incoming data from the in-situ sensors. |

| | |
|---|---|
| **Test procedure** | 1. The user connects to the HEIMDALL VPN.<br>2. The user starts the web portal and logs in.<br>3. The user opens the main page and visualise the map with the different layers.<br>4. The user goes to the in-situ sensors page that displays the location of the sensors.<br>5. The user clicks on one of the sensor icons that opens a window to display the sensor raw data. |
| **Test prerequisites/ configuration** | • The web portal needs to be up and running.<br>• The service platform can serve maps, at least one test site has been configured and information about at least one sensor has been entered in the service platform. |
| **Success criteria** | The user can visualise the following elements on the SP GeoServer:<br>• Maps and layers<br>• Sensors location<br>• Sensor raw information |
| **Results analysis** | In release B, the users could successfully access and visualise the data from the in-situ sensor installed on field. |
| **Success** | PASSED /Tests were successfully accomplished for release B |

Table 6-7: Geotechnical/hydrological monitor test plan. TS_DataSituGeo_2.

| **Test ID** | *TS_ DataSituGeo_2* |
|---|---|
| **Requirements to be verified** | • *TR_DataSituGeo_2*<br>  ○ *Sys_DataSitu_1*<br>• *TR_DataSituGeo_3*<br>  ○ *Sys_DataSitu_1* |
| **Test objective** | The user can visualize the historical and new incoming data from the in-situ sensors. Not only original raw data, but also data derived from operations of original data |
| **Test procedure** | 1. The user connects to the HEIMDALL VPN.<br>2. The user starts the web portal and logs in.<br>3. The user opens the main page and visualise the map with the different layers.<br>4. The user goes to the in-situ sensors page that displays the location of the sensors.<br>5. The user clicks on one of the sensors icons that opens a window to display the sensor modified data. |
| **Test prerequisites/ configuration** | • The web portal needs to be up and running.<br>• The service platform can serve maps, at least one test site has been configured and modified information about at least one sensor has been entered in the service platform. |
| **Success criteria** | The user can visualise the following elements on the SP GeoServer:<br>• Maps and layers<br>• Sensors location |

| | |
|---|---|
| | • Sensor's modified information in engineering units (kPa, cm, %, etc) |
| **Results analysis** | In release B, the users could successfully access and visualise the modified data from the in-situ sensor installed on field. |
| **Success** | PASSED /Tests were successfully accomplished for release B |

Table 6-8: Geotechnical/hydrological monitor test plan. TS_DataSituGeo_3.

| | |
|---|---|
| **Test ID** | *TS_ DataSituGeo_3* |
| **Requirements to be verified** | • *TR_DataSituGeo_4*<br>  ○ *Sys_DataSitu_1* |
| **Test objective** | The user can visualize the historical data from the in-situ sensors. |
| **Test procedure** | 1. The user connects to the HEIMDALL VPN.<br>2. The user starts the web portal and logs in.<br>3. The user opens the main page and visualise the map with the different layers.<br>4. The user goes to the in-situ sensors page that displays the location of the sensors.<br>5. The user clicks on one of the sensors icons that opens a window to display the sensor modified data.<br>6. The user modifies the range of time to be shown by selecting on the GUI the range (4, 8 or 12) and the time unit (hours, days or months). |
| **Test prerequisites/ configuration** | • The web portal needs to be up and running.<br>• The service platform can serve maps, at least one simulation has been started and information about at least one sensor has been entered in the service platform. |
| **Success criteria** | The user can visualise the following elements on the SP GeoServer:<br>• Maps and layers<br>• Sensors location<br>• In-situ sensor information for different ranges of time. |
| **Results analysis** | In release B, the users could successfully access and visualise the simulation results and in-situ sensors information. |
| **Success** | PASSED /Tests were successfully accomplished for release B |

# 7 Conclusion

In this document we provided the technical specifications of two input systems of HEIMDALL: autonomous swarm of drones for monitoring and landslide in-situ sensors based on GB-SAR and Geotechnical/Hydrological monitoring. This document constitutes a reference to guide potential future developers on how to implement such a system, details our implementation, and how our systems interact with the HEIMDALL platform.

# 8 References


[1] Caduff, R., Schlunegger, F., Kos, A., Wiesmann, A., 2015. A review of terrestrial radar interferometry for measuring surface change in the geosciences. Earth Surf. Process. Landf. 40, 208–228.

[2] Monserrat, O., Crosetto, M., Luzi, G., 2014. A review of ground-based SAR Interferometry for deformation measurement. ISPRS J. Photogramm. Remote Sens. 93, 40–48.

[3] Angeli, Maceo-Giovanni, Alessandro Pasuto, and Sandro Silvano. “A Critical Review of Landslide Monitoring Experiences.” Engineering Geology 55, no. 3 (2000): 133–147.

[4] Laporte, Gilbert. "The vehicle routing problem: An overview of exact and approximate algorithms." European journal of operational research 59.3 (1992): 345-358.Kumar, Vijay, and Nathan Michael. "Opportunities and challenges with autonomous micro aerial vehicles." The International Journal of Robotics Research 31.11 (2012): 1279-1291.

[5] R. Murphy, "Fixed- and Rotary-Wing UAVs at Hurricane Katrina," 2006 IEEE International Conference on Robotics and Automation, May 15-19, 2006.

[6] Robin R. Murphy. 2014. Disaster Robotics. The MIT Press.

[7] J. Munoz-Morera, I. Maza, C. J. Fernandez-Aguera, F. Caballero and A. Ollero, “Assembly planning for the construction of structures with multiple UAS equipped with robotic arms”, In Proc. 2015 International Conference on Unmanned Aircraft Systems, 9th June, 2015

[8] Pratt, K. S., Murphy, R. , Stover, S. and Griffin, C. (2009), CONOPS and autonomy recommendations for VTOL small unmanned aerial system based on Hurricane Katrina operations. J. Field Robotics, 26: 636-650.

[9] M. Angermann, M. Frassl and M. Lichtenstern, "Mission review of aerial robotic assessment — Ammunition explosion cyprus 2011," 2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), College Station, TX, 2012, pp. 1-6

[10] BMBF Project ANCHORS: UAV – Assisted Ad Hoc Networks for Crisis Management and Hostile Environment Sensing, Retrieved from https://www.sifo.de/de/anchors-uav---assisted-ad-hoc-networks-for-crisis-management-and-hostile-environment-2135.html

[11] DLR VABENE Project, Retrieved from https://verkehrsforschung.dlr.de/de/projekte/vabene

[12] DLR VABENE Project : Practice test for a UAV Swarm https://verkehrsforschung.dlr.de/de/news/die-multikopter-sind-los-praxistest-fuer-das-swarmtool

[13] Burgard, Wolfram, et al. "Collaborative multi-robot exploration." Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on. Vol. 1. IEEE, 2000.

[14] Lichtenstern, Michael, et al. "Pose and paste—An intuitive interface for remote navigation of a multi-robot system." Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013.

[15] Julian, Kyle D., and Mykel J. Kochenderfer. "Distributed Wildfire Surveillance with Autonomous Aircraft using Deep Reinforcement Learning." arXiv preprint arXiv:1810.04244 (2018).

[16] Merino, Luis, et al. "An unmanned aircraft system for automatic forest fire monitoring and measurement." Journal of Intelligent & Robotic Systems 65.1-4 (2012): 533-548.

[17] Uhlemann, S., A. Smith, J. Chambers, N. Dixon, T. Dijkstra, E. Haslam, P. Meldrum, A. Merritt, D. Gunn, and J. Mackay. “Assessment of Ground-Based Monitoring

Techniques Applied to Landslide Investigations.” Geomorphology 253 (January 2016): 438–51. https://doi.org/10.1016/j.geomorph.2015.10.027.

[18] John Dunnicliff. Geotechnical Instrumentation for Monitoring Field Performance, 1993.

[19] Piksi Multi GNSS Module (https://www.swiftnav.com/piksi-multi)

[20] MCU-30 Lite (https://www.mobilicom.com/mcu-30-lite)

[21] DJI S900 frame (https://www.dji.com/de/spreading-wings-s900)

[22] Pixhawk 2 (http://www.proficnc.com/)

[23] Intel NUC (https://www.intel.es/content/www/es/es/products/boards-kits/nuc.html)

[24] Raspberry Pi (https://www.raspberrypi.org/)

[25] Optris PI400 (https://www.optris.com/thermal-imager-pi400)

[26] USB 2.0 board-level camera - mvBlueFOX-MLC (https://www.matrix-vision.com/USB2.0-single-board-camera-mvbluefox-mlc.html?camera=mvBlueFOX-MLC202bC&selectInterface=Alle&selectMpixels=Alle&selectFps=Alle&selectSensor=Alle&selectColor=Alle&selectSize=Alle&selectShutter=Alle&selectModel=Alle&col=3&row=cmos)Institut Cartogràfic i Geològic de Catalunya, Informe mensual d’auscultació del mes de juliol de 2018 Barberà de la Conca (Conca de Barberà)

[27] RST instruments, Tell-Tale Crack Monitors Brochure (https://www.rstinstruments.com/Brochures/Tell-Tale-Crack-Monitor-EXB0008G.pdf)

[28] Eijkelkamp, Diver: Ground watwer monitoring and communication solutions Brochure (https://diver-water-level-logger.com/download.php?file=2f672b06)

[29] Geokon, Crackmeter Brochure (http://www.geokon.com/content/datasheets/4400_Series_Displacement_Transducers.pdf#page=2)

[30] Geokon, Standard Piezometers (VW) Brochure (http://www.geokon.com/content/datasheets/4500_Series_VW_Piezometers.pdf)

[31] Campbell CR1000 Brochure, (https://s.campbellsci.com/documents/us/product-brochures/s_cr1000.pdf)

[32] Sisgeo datasheet D-Tiltmeter (https://www.sisgeo.com/uploads/schede/schede/S5HD_EN-04_d-tiltmeter_digital_tilmeter.pdf)]

[33] Janeras, M., Jara, J. A., López, F., and Carbonell, T., “Development of a wireless sensor network for rock mass deformation monitoring in the Montserrat Massif,” presented at the 3rd RSS Rock Slope Stability Conference, Lyon, 2016.

[34] A. Viseras, J. Marchal, M. Schaab, J. Pages, L. Estivill " Wildfire Monitoring and Hotspots Detection with Aerial Robots: Measurement Campaign and First Results," 2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Würzburg, Germany, 2019.