Doctoral Dissertation
Doctoral Program in Urban and Regional Development (33rd Cycle)

Open geospatial data and information in support of displaced population contexts

Agata Elia
* * * * * *

Supervisor
Prof. Piero Boccardo, Supervisor

Doctoral Examination Committee:
Prof. Enrico Corrado Borgogno Mondino, Università di Torino, IT
Dr. Mario Angelo Gomarasca, CNR IREA, IT
Prof. Maria Antonia Brovelli, Politecnico di Milano, IT
Prof. Eufemia Tarantino, Politecnico di Bari, IT
Dr. Andrea Ajmar, Politecnico di Torino, IT

Politecnico di Torino
2021

I hereby declare that the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

………………………………....

Agata Elia

Turin, August 2021

Summary

Geoinformation systems and remote sensing proved the extent to which they can significantly contribute to the emergency response. In the context of decision making in emergency situations, high-quality and timely information, accessibility, defined standards, user-friendly interfaces, and appropriate visualisations are all factors of great significance and help.

Displacement settings represent a particularly complex case of emergency. The magnitude of displacement is rising, and nowadays it represents a global theme, involving both developed and developing countries. Displaced people's camps and settlements are getting more numerous and more crowded and the tendency of a high fraction of them is to lose their temporary nature to become permanent and extensive. Managing displacement settings and providing humanitarian support is crucial.

At the same time, the geospatial world is expanding. The availability of open data and free and open source tools is vast, and the capacities made freely available to extract information from it are growing, starting to be based on cloud infrastructures.

The starting point of the proposed research is the impression of an existing gap between available data and the operational use of it. The amount of data generated is impressive and its quality has proven to be good or high, embedding characteristics such as timeliness and accessibility. The majority of remote sensing applications in the field of displaced populations recurs to very high-resolution imagery, whose cost is not negligible and whose temporal resolution and spatial coverage is lower compared to high-resolution imagery.

From field experience it has also emerged that the information extraction task for operational use is generally left to the end user, who sometimes might not have the ability or the resources to use or make best use out of the available data.

Hence, the research questions arise. Is it possible to apply in the context of displacement standardized, operationally focused and simple (relatively to the end user) methods to take advantage of the large quantity of data and of the most recent processing capacities available? Can high-resolution open imagery be implemented in applications supporting displacement contexts? What are the potential applications of high-resolution open imagery in the context of displacement?

Keywords: GIS, remote sensing, Sentinel-2, emergency, IDP, refugees, displacement, DIAS

Acknowledgments

A me stessa,
la montagna
e la musica.

Table of Contents

List of Tables

List of Figures

Chapter 1

1. Introduction

1.1. Displacement as a global event

Natural disasters, including climate change induced disasters, and conflicts are increasingly affecting population worldwide (UNHCR, 2019a; UNHCR, 2018). Displacement is inevitably arising consequently (UNHCR, 2019; Mastrojeni and Pasini, 2017). Forced displacement is expected to increase over the course of the century (UNHCR, 2019). Worldwide, we currently witness 68.5 million of forcibly displaced people, of which 40 million Internally Displaced People (IDP), 25.4 million refugees and 3.1 asylum-seekers (UNHCR, 2019). Displacement will exacerbate the factors leading to conflicts (UNHCR, 2019a), developing a negative self-impacting feedback. Camps are the form of settlement where displaced people (either IDP and refugees) may end up concentrating and residing (WHO, 2019; UNHCR, 2015).

The amount of news that is possible to read every day about displaced people camps is an indicator of the magnitude of the event. From the world's largest refugee camp in Uganda, Bidibidi camp, hosting almost 230,000 people (The Guardian, 2018), to the Syrian Za'atari (The Guardian, 2015) and Bangladesh's Rohingya camps (The Guardian, 2018a), recently threated and heavily affected by heavy rains and floods. It is not only a developing world theme, though, since displaced camps can commonly be set up in the aftermaths of any common emergency, such as an earthquake, like it happened for the earthquakes that hit Italy (L'Espresso, 2009). Displaced people camps are getting more and more numerous and crowded. The tendency for many of these camps, especially in developing countries, is to lose their temporary nature to become permanent and extensive, in terms of permanence and amount of hosted population. Managing these camps, monitoring, and improving the social, sanitary, and environmental condition of hosted population is the key focus of humanitarian agencies and organizations.

Displacement is lately becoming topic of research (Oloruntoba et al., 2018) and it keeps growing as an operational and technical concern of humanitarian agencies and organizations, together with the factors exacerbating it such as climate change. A new branch of research is focusing on the care of displaced people. As an example, the research in humanitarian logistics is recently opening to the theme of IDP and refugees (Oloruntoba et al., 2018). The field of humanitarian logistics was defined by Thomas and Kopczak in 2005 as "the process of planning, implementing and controlling the efficient, cost-effective flow and storage of goods and materials, as well as related information, from the point of origin to the point of consumption for the purpose of alleviating the suffering of vulnerable people" (Thomas et al., 2005).

The main motivation of this research is what is thought to be a growing need for optimized procedures and tools in the world of information and data in support of emergencies. It is well known how timely, accurate, standardized, and accessible data and information are crucial in emergency settings (Konecny et al., 2010), and geospatial data and satellite-based observation are increasingly used to respond to emergencies (Voigt et al., 2016). Most recent emergency events have also exposed the limits of the humanitarian community in coordinating and efficiently use information on most urgent needs of affected population in emergency contexts (Westrope et al., 2013). The planning, management and maintenance of a displaced people camp can be considered as an emergency, and information and data about the camp are crucial for humanitarian actors to plan response and relief (Füreder et al., 2015).

Displacement is an event that impacts society as a whole. Displacement is considered a cross-countries and multidisciplinary event, into which global efforts already flow, but that still remain challenging, complex and needy of several and dynamic solutions. Furthermore, the humanitarian operations in displaced people camps, starting from logistics operations, are potentially well supported by geospatial applications.

1.2. Geospatial information in emergency contexts

The geospatial world is nowadays vast, inter-disciplinary and continuously revised through new technologies. The production of data and their availability is massive and recent platforms make geospatial data, in terms of tabular, vector data or satellite imagery, continuously freely available and updated. Crowdsourcing communities such as OpenStreetMap (OSM) produce and update a detailed map of the world and respond to most emergencies via their Humanitarian OpenStreetMap Team (HOT) (Humanitarian OpenStreetMap Team, 2019). The most recent cloud-based geospatial processing platforms, such as Google Earth engine (GEE), include

a vast database of freely available satellite imagery and geospatial datasets and the possibility to run geospatial analysis on a cloud infrastructure (Google Earth Engine, 2019). The latest developments in the geospatial world include applications of machine learning, training models on imagery and existing geospatial data with the aim of, among others, making predictions for areas where no data is available (MVAM, 2019) or automatically extracting features of interest from satellite imagery (Bramhe et al., 2018).

The vertical applications of produced data and available tools are likewise vast, and an already quite extensive literature about applications for supporting humanitarian operations in displaced people camps exists. Population monitoring, groundwater exploration, environmental impact assessment or land degradation (Füreder et al., 2015) are few examples, as they are going to be covered in the section framing the research topic.

The starting point of the proposed research is the impression of an existing gap between available data and the operational use of it. The amount of data generated is impressive and its quality has proven to be good or high, for instance for crowdsourced OpenStreetMap data (Minghini et al., 2018). Open data, in addition, embed quality such as timeliness and accessibility (Elia et al., 2018), which are considered to be useful in the context of displaced people camps (Füreder et al., 2015). Furthermore, from the field experience briefly exposed in the next section, it has emerged that the operational information extraction task is generally left to the end user, who sometimes might not have the ability or the resources to use or make best use out of the available data.

1.3.   A field experience

A field experience parallel and connected to the topic of this research was carried out by the author in 2019 in Uganda with the REACH IMPACT (https://www.impact-initiatives.org/) NGO, based in Kampala. This experience offered the possibility to experience actual field issues connected with geospatial information in displacement contexts. The NGO's main activity, indeed, is to facilitate the development of information tools and products that enhance the humanitarian community's decision-making and planning capacity. In Uganda, this NGO mainly works in performing assessments related to different sectors such as Multi Sectors Needs Assessment, House Land and Property, and others, in all 30 existing refugee settlements and in urban refugee locations, such as Kampala (IMPACT, 2021).

Thanks also to this field experience, it was possible to understand current best practices in the field of data collection and data management in support to displaced people camps. In addition, a vision on potential gaps and room for improvement in existing methodologies and where new methodologies could be explored in support

to humanitarian operations in displaced people camps was acquired. The operational use of available data and the limits of a team in accessing, understanding, and extracting relevant information from the data when needed was experienced. How a needs assessment is performed and how geospatial methods can support it was a crucial aspect of this field experience.

Key take away points of this field experiences were:

- The evidence of the existing room for automation of processes in optimizing the information flow in displacement context related emergencies.

- The evidence that the operational information extraction task is generally left to the end user, who sometimes might not have the ability or the resources to use or make best use out of the available data.

1.4. Research questions and objectives

The main objective of the research is in first place to explore and investigate the applications of geospatial tools and methods in support of emergency settings. In particular, understanding how data and information, particularly the geospatial component of them, fits in the context of humanitarian operations in displaced people camps or settlements. After exploring and evaluating current best-practices in the field of data collection and management in support of displaced people camps, the aim is to identify potential research gaps. The ultimate objective is to make a step toward the varying and complex information needs of the humanitarian displacement contexts, possibly bridging the large amount of open data and free and open source processing capacities available with existing standardized operational needs. The importance and the use of the geospatial component of data and information has increased over the past years in the emergency sector. This research is expected to explore and investigate applications of geospatial technologies in support of emergency settings, particularly in the complex context of displaced people camps and settlements.

1.4.1. Specific research questions

How is the geospatial technology currently supporting humanitarian operations in displacement settings? What geospatial/non-geospatial method or combination of methods currently meet information needs in displacement settings (remote sensing, drones, GPSs, databases, platforms, and apps, etc.)? What are the needs in terms of information in displacement contexts? Is there room for automation of processes to obtain higher efficiency of data? Is the large availability of data in this context optimally used and if not, how can this be optimized? Is it possible to apply in the context of displacement standardized, operationally focused and simple (relatively to the end user) methods to take advantage of the large quantity of data

and of the most recent processing capacities available? Can high-resolution open imagery be implemented in applications to displacement contexts? What are the potential applications of high-resolution open imagery in the context of displacement?

1.4.2. Aim and contribution of the research

The general aim of this research is a potential contribution in exploitation of available technologies and data with an operational use objective. This bridging the context and research in humanitarian operation for displacement settings with the research in geospatial methods and tools, having in mind standards and real needs in humanitarian and emergency operations. Strongly based on real case studies and observations, the research is aimed at testing and exploring different available tools and in understanding limitations and potentials.

Literature and operational real cases examples on applications of geospatial methods for the care of displaced people exist. Anyhow, the technologies are moving fast, and more and more data are made freely available as well as processing capacities. Understanding whether available open data and the new open-source and cloud-based processing capacities can contribute to the emergency management sector and particularly to the displacement settings context is also one of the expected contributions.

1.4.3. Target subjects

The main domain of interest of the research is the humanitarian and emergency management domain. Targets can be humanitarian actors as well as other researchers in the attempt to fit in a wider network of studies. Once the results of the research would be available, it might encourage humanitarians to feed their data or to give suggestion on type of analyses that can be performed or methods that can be applied.

1.5. Methodology and expected results

The starting point of the research is a review of existing academic literature and operational real cases about applications of geospatial methods for operations in displacement settings in order to frame the research topic. Combining the keywords GIS, remote sensing, satellite imagery, data, information, refugees, IDP, refugee camps, refugee settlements and informal settlements, the resulting literature is reviewed. This will allow to categorize the main technologies already applied in the context of displacement settings and to identify main applications categories as well as specific applications. Pro, cons, and limits of already applied technologies will possibly be highlighted as well as specific requirements in terms of data of each application category.

Following, by further investigating the use of the already listed technologies in applications for displacement settings with a focus on automation, the aim is to possibly tackle existing gaps in existing applications and to understand whether it is possible to make use of the most recent available technologies and released data to fill these gaps.

Afterward, a series of potential applications in the context of displacement of open data and different existing tools and methods will be explored, and their results analysed and presented. The following scheme sums up the research schema followed.
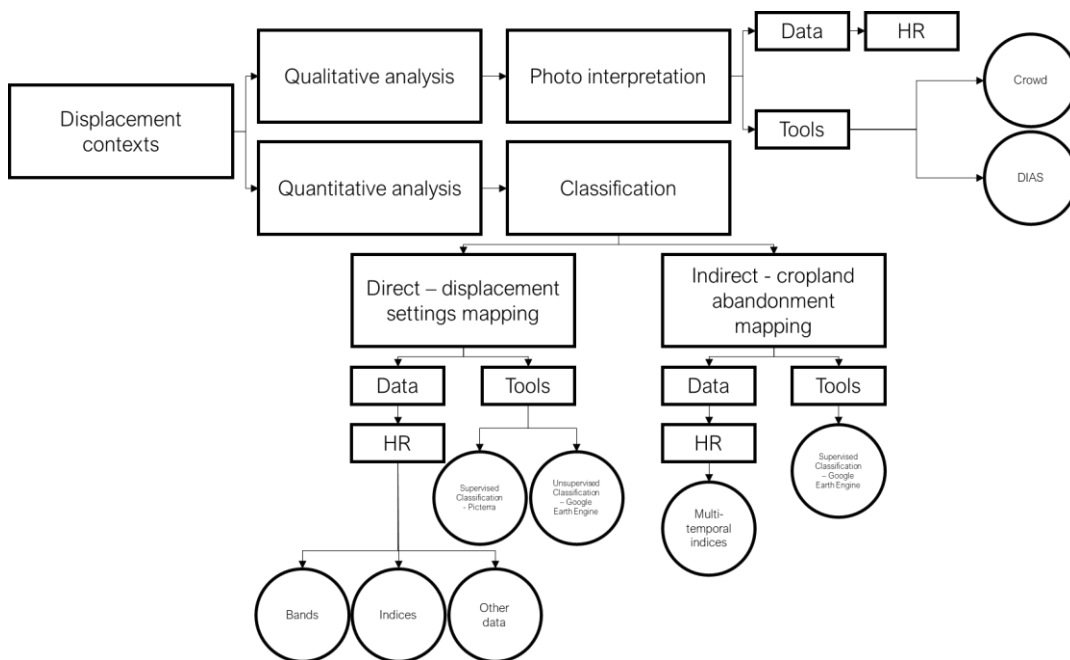
Figure 1.1 Research schema.

The results expected from this research are:

- An evaluation of how geospatial methods and technologies can effectively and efficiently support humanitarian operations in displacement settings, looking at already explored methodologies in the literature and especially looking at the newest available geospatial capacities.

- A categorization of which methods and technologies provide support and the identification of their benefits and drawbacks.

- The identification of a research gap in the field of geospatial methods applied to displacement context.

- A series of examples of how geospatial open data and newest available geospatial technologies could be used in the context of displacement and their results.

1.6. Framing of the research topic

The growing research interest in the field of displacement, particularly supported by geospatial and remotely sensed information, is highlighted in the following charts representing the research documents by year identified by the following queries within the Scopus database (Scopus, 2021):

- ((TITLE-ABS-KEY(((remote sensing) OR (satellite imagery)) AND ((refugees) OR (displaced W/0 people) OR (displaced W/0 population) OR (refugee W/0 camps) OR (refugee W/0 settlements))))). With a total of 159 results.
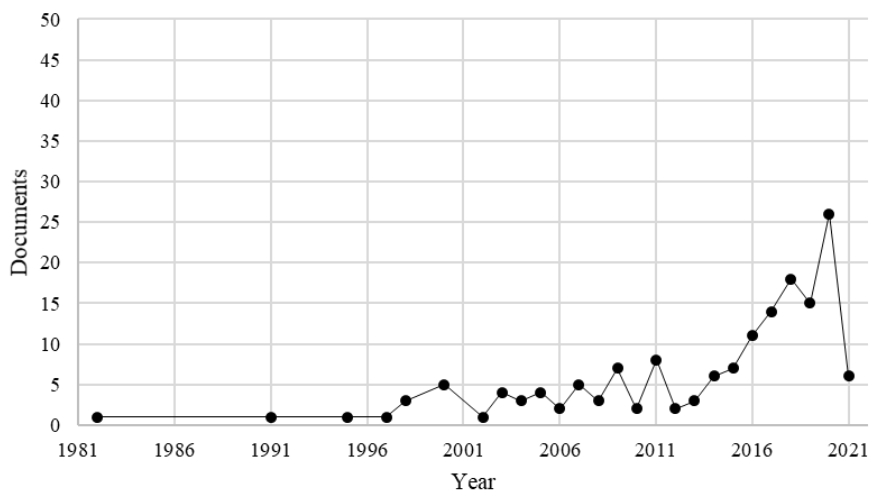
Figure 1.2 Trends of scientific production.

- ((TITLE-ABS-KEY(((remote sensing) OR (satellite imagery)) AND ((refugees) OR (displaced W/0 people) OR (displaced W/0 population) OR (refugee W/0 camps) OR (refugee W/0 settlements) OR (informal W/0 settlements))))). With a total of 330 results.
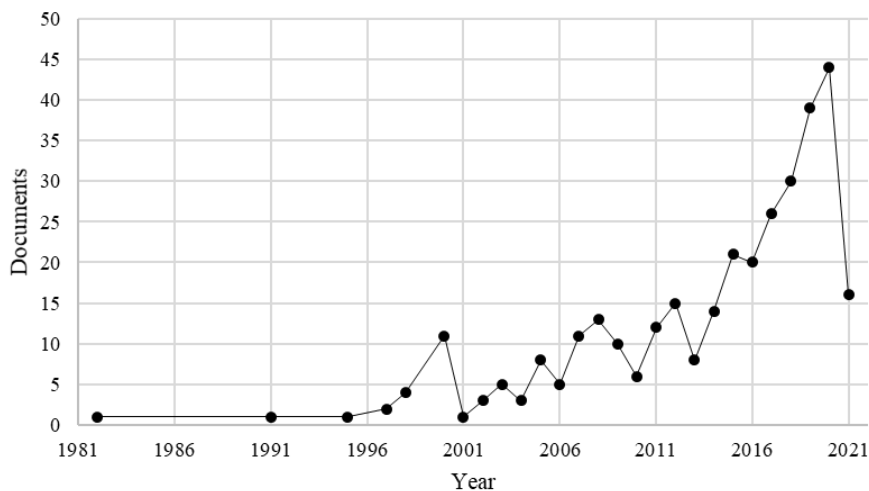
Figure 1.3 Trends of scientific production.

In the emergency context, a displaced person is defined as someone who had to abandon her/his/their home due to natural, technological, or deliberate event (WHO, 2019). Displaced people is a broader definition that includes Internally Displaced People (IDP) as well as refugees. IDP are people who remain in their own country after displacement, while refugee are people who cross international borders as a result of displacement (WHO, 2019). Camps are the form of settlement where displaced people (either IDP and refugees) may end up concentrating and residing (WHO, 2019; UNHCR, 2015). Camps can widely vary in size, quality, type of equipment and location according to the funding the camp receives and on the hosting country's policy in the matter (Schön et al., 2018). The main characteristic that defines a displaced people camps is that they are intended to be temporary (USA for UNHCR, 2019). Camps allow the hosted population to receive formal centralised protection, humanitarian assistance and other services from host governments or other humanitarian actors (UNHCR, 2015). When the emergency evolves into a protracted crisis, the camps may lose their provisional nature and turn into long term settlements (USA for UNHCR, 2019). The protracted nature of a settlement generates new challenges that include delivering of infrastructure, energy, and education to what slowly become a community (USA for UNHCR, 2019). An estimate of 14.2 million people resides in camps for an average of 17 years (Schön et al., 2018). Finally, when the camp or settlement is populated spontaneously without the assistance and guidance of local government and humanitarian community, the camp or settlement is informal (UNHCR, 2015a) and the provision of humanitarian aid is not officially established anymore. Food, water, sanitation, health, and education assistance are not formally established, and physical shelters are sub-standard (REACH, 2013). The arrival of displaced people generates critical issues in terms of shelters, health, food security, nutrition, education, environment, energy, and protection, concerning both the displaced people themselves but also the communities hosting them (FAO/UNHCR, 2016).

Geoinformation systems and remote sensing have already proven several times the extent to which they can give a significant contribute to the emergency response (Konecny et al., 2010). Over the last years, geospatial technologies and earth observations have supported humanitarian action, specifically logistics in planning and mitigation in the aftermaths of natural disasters or complex crisis (Füreder et al., 2015). In the context of decision making in emergency situations, high-quality and timely information, accessibility, defined standards, user-friendly interfaces, and appropriate visualisations are all factors of great significance and help. This is where geospatial methods and tools allow improving user perception, orientation, knowledge and understanding. Furthermore, digital geospatial data include the relevant potential capability of adaptation of their visualization with respect to the user and of being queryable, an aspect of great significance considering the multi-stakeholders implication aspect of an emergency (humanitarians, NGOs, media, local communities). All these concepts are especially valid nowadays, where digital cartography created the shift to ubiquitous mapping and the capability of acquiring and supplying enormous amounts of geoinformation and, at the same time, the

maintenance and accessibility of spatial data infrastructures, has greatly improved (Konecny et al., 2010).

If the usefulness of Geographic Information Systems and emergency data is taken for granted, still the shortcomings and failures of existing technologies and applications in emergency response are equally known. Many barriers still exist in making data available, appropriate, and reliable and in making systems and providers working together and in coordination, avoiding overlays in products and time wastage. The importance of enhancing data availability and information sharing is a crucial aspect in the definition of the actual potential of emergency mapping in the Disaster Risk Management context. Furthermore, data have not only to be displayed, but they have to be made readable avoiding overwhelming amount of undefined information and be reliable. Finally, cartographic products are highly technical and still require the skills of trained and educated experts to be produced and interpreted (Harvard Humanitarian Initiative, 2011).

Overall, most recent crisis and disasters have exposed the limits of the humanitarian community to coordinate a rapid gathering and an effective use of information, both already existing and required in the aftermaths of the crisis, leading to significant information gaps. Two main gaps are highlighted. The first one concerns the emergency phase of the crisis, when the data supply is lower compared to the demands. The second one arises in the recovery and development phase. When the available information exceeds the demands, the use of this information decreases with a subsequent decline in the regular supply of information products, leading to an information gap for an eventual future occurring crisis (Westrope et al., 2013).

In the case of displacement settings considered as a complex emergency, having as a starting point geospatial data and satellite or aerial imagery, critical information can be produced and provided for humanitarian operations (Füreder et al., 2015). Information management is one of the Camp Coordination and Camp Management (CCCM) key concerns in implementing the response. According to the UNHCR Emergency Handbook, a centralized information management system should be developed, disseminating information management products regularly. Particularly in the initial stages of an emergency, the harmonization of information is relevant for efficient sharing (UNHCR, 2015b). A solid information management ensures that all actors involved in the emergency are working with same or complementary relevant, accurate and timely information and baseline data. The main information management phases highlighted are data and information collection, data analysis and information dissemination (CM Toolkit, 2015). The Camp Management (CM) CM Toolkit by the CCCM Cluster highlights the following main information needs at camp level:
- Information on total number of camp inhabitants and their status (refugee/IDP/stateless), their age, sex and protection need;
- Information about groups and individuals most at risk;

- Information about services, standards and activities in the camp;
- Information on camp security and safety;
- Information on procedures and systems in camp management;
- Information on camp infrastructure: roads, pathways, communal buildings, health centres, schools, distribution sites, latrines, drains, water supply lines, electricity lines, meeting places and burial sites;
- Information on coordination mechanisms: Who is doing what, where (3W) (CM Toolkit, ,2015).

So far, a wide range of geospatial methods and tools has been already deployed to support displaced people camps and has been documented in research. A camp's life cycle is generally described in three phases: set-up, care and maintenance and closure (CM Toolkit, 2015). The different phases of a displaced people camp's life cycle involve different information needs and consequently different methods and tools that can be applied in this context. After a literature review of available papers and articles covering geospatial applications in displacement settings, the deployed methods and tools can be summarized mainly in the following technology categories:

- Satellite imagery. Satellite imagery can be acquired for non-accessible areas all over the world. The temporal resolution of satellite imagery allows potentially both past and real-time investigations. On the other hand, spatial and temporal resolution varies greatly with available budget and environmental factors (e.g. clouds, atmosphere).
- UAVs/Drone. Unmanned Aerial Vehicles use is expanding in a wide range of sectors including the humanitarian operations. The possibility to target users' needs in terms of spatial and temporal resolution of acquisitions, area coverage and to bypass adverse climatic conditions such as cloud coverage are crucial in the humanitarian sector. In addition, a portable technology serves well community mapping projects. Problems arise in terms of regulations, including security and privacy concerns (FloodList, 2016).
- Field collected geospatial data (GPS based surveys). The most direct application of geospatial methods is the acquisition of the GPS location of features of interest on the ground, with the possibility to collect and store directly observed information and details attached to the GPS location. Several mobile data collection applications are nowadays available, and their use is common and standard across humanitarian organizations, recurring mostly to Kobo and ODK applications. Unfortunately, in emergency contexts, it is not always possible to access locations directly. Furthermore, complete surveying of infrastructures is time and resources consuming and it becomes even more complex when the context is highly dynamic and changing quickly, such in the case of displaced people camps.
- Databases. Not an exclusively geospatial method. The possibility to store efficiently and effectively information is crucial in emergency contexts as well in displacement settings. A standardized, unified, accessible and

23

queryable information management system is helpful in supporting humanitarian operations. Given the vast deployment of different actors in the humanitarian response, effectively set and use standards across them to collect and update information is highly challenging.

Each of these technologies have been used individually or simultaneously and have proved to have pro and cons in the specific displacement settings sector. Main categories of applications reviewed so far through the literature review process are summarized below. Sources are both academic papers and articles covering real operational case studies.

- Population and infrastructure mapping and monitoring.
  Very high-resolution (VHR) satellite imagery has been used to derive information on the amount and spatial distribution of shelters in displacement settings. Information about the location, size and hosted population of a camp is fundamental to plan first response but also long-term humanitarian aid. When no information is available on the ground, satellite imagery can provide a first estimate of location and size of the camps (Füreder et al., 2015). When enough and reliable field information is available, this data can be used as a proxy for the population residing in the camp (Kemper et al., 2011). Being camps highly dynamic contexts, satellite imagery is also implemented to monitor movement, expansion, or reduction of the camps (Copernicus, 2018; Kemper et al., 2011). Shelter mapping is performed via visual interpretation of imagery mostly, but the literature on shelter mapping is very wide and is moving toward the semi-automatic and automatic extraction of shelter footprints. The issue of automatic shelter mapping is very complex and presents several obstacles such as the need for VHR imagery, the difficulties in differentiating between displaced people shelters and host community shelters in peculiar and complex structural context, where the displacement setting is not constituted by tents and complex spectral contexts, with high shelter density, low shelter contrast or different shelters materials. The papers available in this context, in fact, present automatic extraction of camp shelters and camp extent for very specific case studies and locations, since generalization is highly challenging (Lang et al., 2010). Object-based image analysis (OBIA) is used to distinguish objects based on their shape, size, spectral behaviour, and neighbouring relations (Füreder et al., 2015). Anyhow, the limits of semi-automated tools in the humanitarian domain are well recognized, given the complexity of the features of interest (Voigt et al., 2014).

- Natural resources assessment.
  Natural resources are critical in displacement settings. Issues such as access to fuel and energy, environmental degradation around camps, availability of water resources are all crucial aspects in displacement contexts

(FAO/UNHCR, 2016). The settling of a large number of displaced people into an area has proven to induce a stress on the surrounding environment, leading to impacts on forests, land and water resources (Füreder et al., 2015). Access to safe and reliable sources of fuel and energy for cooking, lighting and other uses is essential for sustainable development and food security of displaced people. The main fuel used in displacement settings (in developing countries) is wood, which is used to produce charcoal. This has become one of the main drivers of environmental degradation around the camps sites. Remote sensing has been used to assess wood fuel supply and environmental degradation of targeted areas. Time-series of appropriate satellite imagery have been analysed to detect different land cover types significant for wood fuel production, mainly tree cover and shrub cover, and their changes over time. In connection to this, field measurements allowed to derive biomass expansion factors from satellite imagery to estimate biomass stocks (FAO/UNHCR, 2016). The availability and supply of drinking water, in a reasonably close range, is as well critical in displacement settings, especially in the long term when groundwater takes the primary role of water provision when purification of surface water and water trucking are not sustainable anymore. Remote sensing has been used to define suitable groundwater extraction sites, making an effort in delineating the location and extend of subsurface structures potentially containing water (Wendt et al., 2015).

- Risk or damage assessment.
  Displacement settings often take place on dangerous, hazard-prone sites such as hill-slopes and flood-prone areas. In some cases, the hazard is generated by the displaced people themselves, by over-exploiting natural resources, for instance cutting hills and destroying forests for building shelters and providing fuel, like in the case of the Rohingya refugees in Bangladesh (Sammonds et al., 2018). Displacement settings become highly vulnerable to heavy rainfall causing severe flooding and shelters, infrastructure and livelihood destructions, water sources contamination and further displacement, such in the latest floods in northern Syria (Reliefweb, 2018) and in Cox's Bazar, Bangladesh (REACH, 2018a). Flood and landslide risk mapping are some of the examples of applications of geospatial methods for risk or damage assessment in displacement settings. Using elements such as elevation, water flow directions and previous rainy season observations it is possible to assess flood-prone areas and identify infrastructures likely to be affected by floods and landslides (REACH, 2015). The use of UAVs/drones imagery has also proved to be useful in flood risk mapping exercises in urban informal areas, whose vulnerability to extreme events resembles the one of displacement settings. UAVs/drones provide very high-resolution and up-to-date imagery targeted to the user needs in terms of resolution, acquisition time and area coverage in addition to the possibility to bypass adverse climatic conditions such as cloud

coverage, characteristics that serve well flood risk and exposure mapping. These characteristics are hardly reachable recurring to satellite imagery. In addition to allowing the generation of a 3D surface model of an area and potentially a detailed hydrological model, very high-resolution (5 cm) drone mapping allows detailed digitalization of features, hence high-quality mapping of exposed areas (Open DRI, 2018; FloodList, 2016). Extent of actually flooded areas and damage in the aftermaths of an extreme event can then be extracted recurring to very high-resolution radar imagery and semi-automatic processing techniques (UNITAR, 2018; Copernicus, 2013).

Figure 1.4 Figure 1.2 Zaatari camp flood map (Copernicus, 2013).

- Camp site selection.

Facilities location in emergency settings is one of the themes faced in the humanitarian logistics sector and that can be supported by geographical information systems (Rodríguez-Espíndola et al., 2016). Models are generated based on vector and raster inputs and on a set of rules to obtain optimisation-based disaster management systems to locate facilities or identify optimal routes in case of emergency (Rodríguez-Espíndola et al., 2016). Same theory can be applied to displaced people camp site selection (Çetinkaya et al., 2016). The location of the camp has a remarkable impact on access to assistance, protection, and well-being of hosted population throughout the camp life cycle. In many cases displaced people settle spontaneously before aid is available, occupying unsuitable available and unwanted land, often prone to natural hazards such as floods and with inadequate supply of basic services such as WASH (water, sanitation and hygiene) and health facilities (UNHCR, 2015).

- Field data collection.

The most straightforward implementation of GIS technologies is in the planning, implementation, monitoring and post-processing of field data collection or more generally of field surveying. The response in displacement settings is planned according to the needs of the people residing in it. Usually, these needs are firstly estimated by obtaining basic information about the camp (location, extension, number of structures and consequent estimated number of people) also through remotely sensed data, as shown in the previous paragraphs. In a second phase, information is collected through surveys, to identify more specific and population-oriented needs for better resources allocation and humanitarian interventions. Needs assessment is one of the areas identified as needy to be improved by the agreement among major humanitarian donors in 2016, the Grand Bargain, mainly because of the lack of a standardized and coordinated information gathering and analysis system. The main aim of needs assessments is to fill information gaps of organizations and agencies leading the humanitarian response (Agenda for Humanity, 2016). Needs assessments are carried out through data collection campaigns, namely surveys. Survey can be statistically representative, hence surveying a statistically relevant sample, or they can cover the whole population, becoming a census. Surveys are designed with tools in different surveying applications, mostly Kobo and ODK in the humanitarian sector, and they are carried out with mobile data collection applications. In this, geospatial methods are relevant both in the planning of the survey, addressing locations needed to be surveyed, and in the data collection, where the GPS locations of the surveyed households (in case of a household level survey) is registered and represents a precious information to perform further analysis on the final obtained database of surveyed households (REACH, 2018). A limit in statistical representative surveys is that, according to the resources available, the final survey can be representative only at whole camp level or in optimal cases at camp zones level, limiting the spatial and statistical analysis applications. In efforts of extensively surveying and keeping track of a displaced people camp, meaning that all households of a camp are included in the survey, it was possible to build comprehensive camp level queryable databases. In an existing operational example in South Sudan, each household is identified by a unique identifier that allows to link information on that household from disparate sources and link them into a single database, including its GPS location. This allows the provision of targeted georeferenced aid and a more accurate and potentially up to date planning and coordination of the camp. The main effort is to maintain the efficiency of such a database by standardizing information formats across partners, so that the database can be kept updated with on-going data collection campaigns. In addition, being a displacement setting highly dynamic, the update of such a database needs to be regular and frequent. Embedding the geospatial component in a camp database allows to spatialize information and display it, allowing for easier

information sharing and analysis capacity including each feature of the camp. (Westrope et al., 2013).
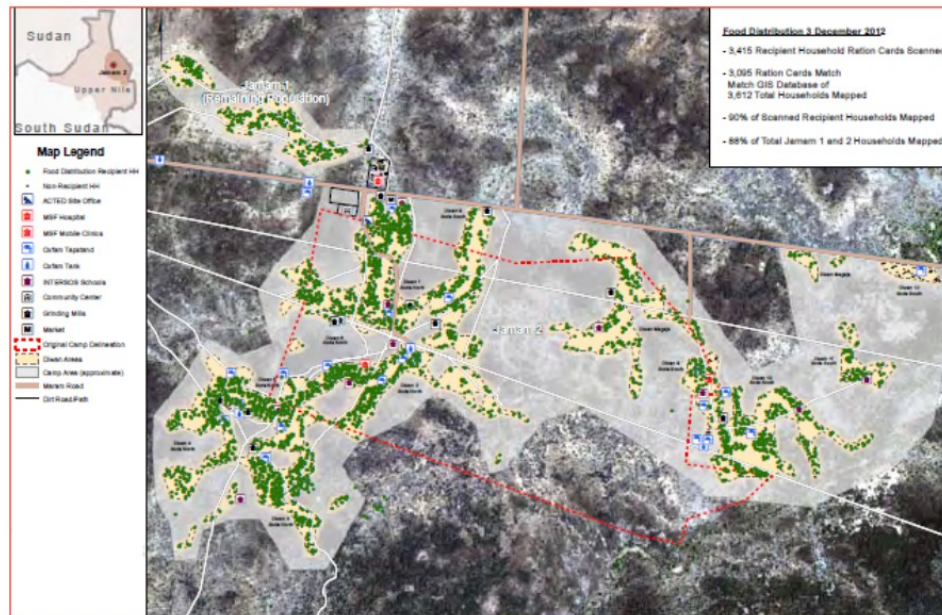
Figure 1.5 Map of Jamam camp food distribution recipients and non-recipients (Westrope et al., 2013).

In addition to this, the field surveying of infrastructures is another precious resource where geospatial methods play an important role. Mapping infrastructures GPS locations and details using mobile data collection applications with GPS embedded devices, allows conducting surveys of existing facilities and infrastructures in the camp, whether WASH, health, education, or whatsoever. This is helpful in identifying coverage, functioning and vulnerability of existing infrastructures and their ability to serve the current population (Reliefweb, 2017).
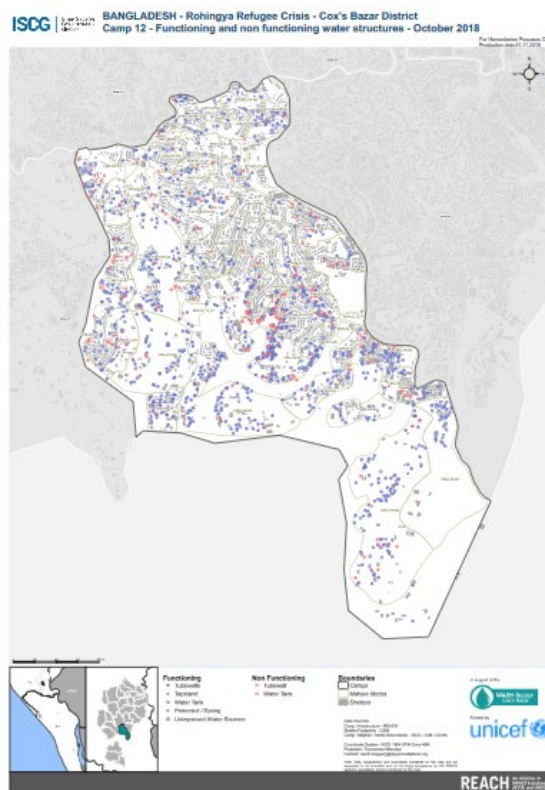
Figure 1.6 Functioning and non-functioning water structures in Kutupalong Camp 12 (Reliefweb, 2017).

The following table summarizes the major findings of the performed literature review. Classes of spatial resolution refer to ESA classification of resolution of sensor of their missions (ESA, 2015):

- VHR1 Very High Resolution 1 where resolution <=1m
- VHR2 Very High Resolution 2 where 1m < resolution <=4m
- HR1 High Resolution 1 where 4m <resolution<=10m
- HR2 High Resolution 2 where 10m <resolution<=30m
- MR1 Medium Resolution 1 where 30m <resolution<=100m
- MR2 Medium Resolution 2 where 100m <resolution<=300m
- LR Low Resolution where resolution >=300m

Application category	Application	Tools and methods	Spatial resolution (if applicable)	Reference
Population and infrastructure mapping and monitoring	Shelters mapping	Remote sensing Drones	Very high	Füreder et al., 2015 Kemper et al., 2011 Copernicus, 2018 FloodList, 2016
Natural resources assessment	Delineation of potential groundwater extraction sites	Remote sensing	High	Wendt et al., 2015
	Wood fuel supply and environmental degradation	Remote sensing	High Very high	FAO/UNHCR, 2016 Füreder et al., 2015
Risk or damage assessment	Risk assessment	Remote sensing Drones	Very high	REACH, 2018a Open DRI, 2018 FloodList, 2016
	Flood mapping	Remote sensing	Very high	UNITAR, 2018 Copernicus, 2013
Camp site selection	--	Remote sensing	High	Rodríguez-Espíndola et al., 2016 Çetinkaya et al., 2016
Field data collection	Needs assessment	Databases GPSs	--	REACH, 2018 Westrope et al., 2013
	Infrastructure mapping	Databases	--	Reliefweb, 2017

		GPSs		

The research in the field of displacement settings and in finding solutions for the care of the people residing in it is being highlighted for its potential and pushed to be investigated further, especially in the humanitarian logistics sector (Oloruntoba, 2018; Seifert, 2018). The interest of this proposed research is to bridge the growing interest in this field with the geospatial research. By further investigating the use of the already listed technologies in applications for displacement settings, the aim is to possibly tackle existing gaps and to understand whether it is possible to make use of the most recent available technologies. No scientific paper has been found so far, for instance, on applications of Google Earth Engine (GEE) or of more general Data and Information Access Services (DIAS), such as the ones provided by the European Commissions (Copernicus, 2019), in the field of displacement settings. These cloud-based platforms provide access to data and information and to processing tools to directly analyse them, bringing availability and accessibility of data to another level. The database of available satellite, vector and tabular data is vast in Google Earth Engine (Google Earth Engine, 2019a) and the European Commission is including in his newly funded DIAS also all the information products deriving from its operational services (Copernicus, 2019) such as the Emergency Mapping Service, that often produced information on displacement settings digitizing it from VHR satellite imagery (Copernicus, 2019a). As highlighted in the summarizing table, the biggest challenge for using these platforms seems to be the resolution of available imagery. While the availability of data is huge and potentially covering thematic and temporal needs of the applications illustrated in the summarizing table, the resolution of freely available imagery is mostly high or medium, while applications in small scale context such as displacement settings are mostly supported by very high-resolution imagery, which is mostly not freely available and needs to be purchased. GEE, for instance, includes a free repository of over 40 years of global satellite imagery at high temporal resolution, from multiple satellite missions including, among many others, the complete Landsat series, Moderate Resolution Imaging Spectrometer (MODIS), National Oceanographic and Atmospheric Administration Advanced very high-resolution radiometer (NOAA AVHRR), Sentinel 1, 2, and 3, Advanced Land Observing Satellite (ALOS). Minimum spatial resolution of available imagery in GEE is of 10 meters. The largest applications of GEE from 2010 to 2017 were in the natural resources' domain (vegetation, forest, land cover and land use change monitoring) (Kumar et al., 2018). Anyhow, also vector, social, demographic and weather data are included in the platform (Kumar et al., 2018), which might be potentially useful. The cloud processing capacity of these systems could be also further investigated to support information production on displacement settings. For instance, recurring to machine learning algorithms in a cloud computing environment for training classifiers powered by the wide range of spectral, spatial and temporal resolutions available, or even just importing secondary sourced raster or vector datasets to perform targeted analysis or making use of the GUI to make

GEE scripts into web applications (Kumar et al., 2018). In addition to this, studies on extractions of settlement extent and extent change from Landsat imagery (high-medium resolution) using Goole Earth Engine exists (Patel et al., 2014), and this might be further investigated for applications in the larger and denser displacement settings.

Optimally visualize and efficiently extract information from existing datasets is another interesting field of research, addressing issues such as usability and transferability of data. A successful example of a similar application is ADAM, the Automatic Disaster Analysis and Mapping system of the World Food Programme developed with open-source technology. This alert system automatically retrieves information from a series of sources (Global Disaster Alert and Coordination System, the US Geological Survey, World Bank, WFP databases) before, during and after an earthquake or a tropical storm happen. The alert-system then produces real-time virtual dashboards with event relevant information such as scale of the emergency, population likely to be affected, weather conditions, available resources etc. This timely and accurate information allows humanitarian operations to be carried out more informed and to plan preparedness actions accordingly (World Food Programme Insight, 2018).
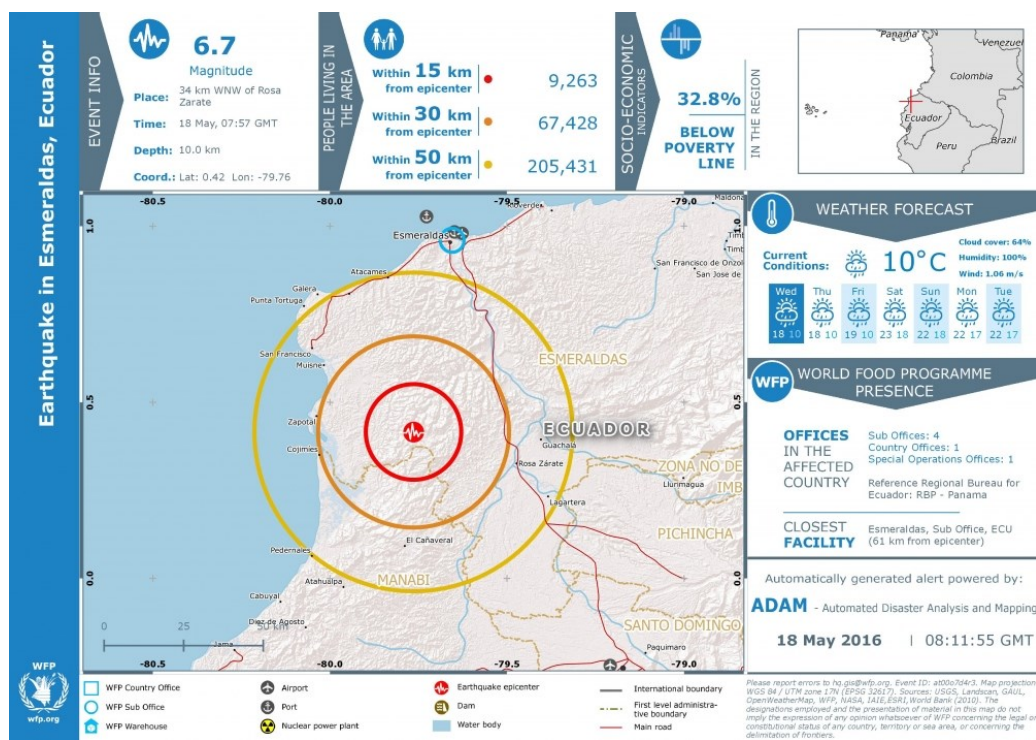
Figure 1.7 ADAM output in the aftermaths of Ecuador earthquake (World Food Programme Insight, 2018).

An attempt to summarize in visually appealing and interactive maps information and data related to displaced people camps exists as well. A project for the Syrian Zaatari camp created a tool to incorporate in an interactive map and in charts the analysis of key social attributes of the camp population, derived from camp specific survey data and displayed on OpenStreetMap data in a bivariate

choropleth map. The aim was to inform needs assessment and resource allocation (Mooney et al., 2017).
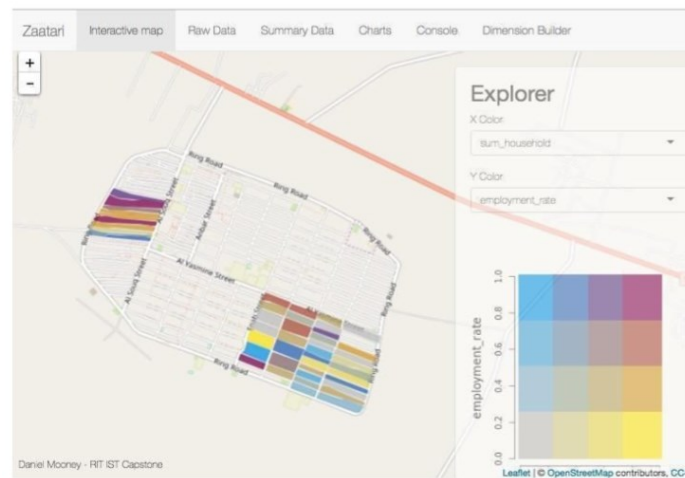
Figure 1.8 Zaatari camp choropleth interactive map (Money et al., 2017).

Nevertheless, no attempt has been found so far to create a more automated and not camp specific tool with methods similar to the ADAM WFP tool, which may be able to pull displacement settings data from existing global sources and produce information on an interactive map or dashboard. Sources of data about displacement settings are nowadays vast, including OpenStreetMap, whose Humanitarian Team constantly work on new humanitarian projects (Humanitarian OpenStreetMap Team, 2019), and the Humanitarian Data Exchange platform, where major organizations, such as UNHCR, WFP, etc., upload their data including facilities locations (WASH, education, health, etc.) and population data (HDX, 2019). Differently from the ADAM WFP tool, this would not be a rapid assessment tool but would need to be built on information collected over time.

In the next Chapter, the framing of the research topic will be continued, and it will be focused particularly on those applications and research dedicated to the automation of satellite imagery analysis for displacement contexts. Finally, the existing identified gap in the scientific literature that will be tackled by the presented research will be framed.

Chapter 2

2. Automation in remote sensing for displacement contexts

Machine learning techniques have been tested widely in the humanitarian context over the last years, including specific exercises on displaced people camps or settlements. As it was illustrated in Chapter 1, major applications of remote sensing for humanitarian purposes in these contexts require skilled human visual interpretation and manual mapping over imagery. Consequently, this framework is particularly suitable to a certain degree of automation. Indeed, automating the detection of specific features, such as camps/settlements or specific camps/settlements structures, can potentially minimize human efforts, shorten the time of response, and align outputs. Even more importantly, automation could potentially amplify the use and the benefits of the increasing availability of satellite imagery. Finally, this would lead to more frequent and more detailed monitoring (Quinn et al., 2018).

What were successful and not successful applications of automatic or semi-automatic detection of these specific features supported by remote sensing so far? The detection and classification of displacement settings form remotely sensed imagery can easily fit in the broader context of urban or human settlements applications. Applications of machine learning for detection of urban areas and human settlements in remotely sensed imagery are extremely extensive and some of that are going to be taken into consideration in the next sections.

The specific sub-case of displaced population camps and settlements is a more matted one. First of all, why one should proceed in the direction of applying machine learning for displacement settings? Some answers are the following:

- As explained in Chapter 1, displacement is a humanitarian growing issue.

- The amount of open data and free and open source tools available nowadays is outstanding and the potential of them is partly unexplored.

Why displacement settings applications are more challenging compared to general human settlements applications, such as urban areas?

- Displacement settings are often temporary, or they exist withing a specific time window. Therefore, training of a classifier should be done at the date when the camp exists. This means different dates per each camp, as different camps may exist and disappear at different dates.

- There is no global updated unified database on locations of displacement settings. This represents a lack of ground truth data compared to high availability of satellite imagery and analytical tools.

- Displaced population camps/settlements are fewer in number compared to urban areas. Consequently, it is very likely to have less training data input compared to general urban areas applications.

- Displacement settings have a peculiar and complex pattern.

- Displacement settings show high variation in patterns and appearance within same feature type according to geographic area, type of event, type of setting, surrounding environment, etc.

- Displaced population camps and settlements may have a high spectral similarity to urban features. This may make difficult to isolate and separate the two type of features.

2.1. Machine learning and deep learning

First applications of machine learning methods in remote sensing have been devoted to land cover classification exercises using HR multi-spectra imagery. First attempts recurred to random forest and support vector machines algorithms. These algorithms have represented a standard for remote sensing applications for about ten years. Lately the applications of machine learning to remotely sensed imagery have shifted towards deep learning methods.

In machine learning the problem is generally split into data labelling, training, testing and validating, deep learning has smoothened the pipeline incorporating the model learning and fine tuning of the model parameters.

The biggest difference between machine and deep learning is the amount of training data and the computational power demanded, which are way larger in deep learning exercises. Common network structures used in deep learning exercises for remote sensing applications are autoencoders, usually used to map the extent of features, hence applicable in land cover mapping exercises, or convolutional neural networks for object detection methods, hence for identifying the bounding box of objects detected in the imagery.

Machine learning and deep learning applications to remote sensing span from high precision land cover mapping to scene classification and so on. Of course, a not negligible aspect is the huge amount of data and the large computational power required to train from scratch a deep learning model on a brand-new problem. This is the reason why usually transfer learning is used as a solution. When low capacity and poor datasets are available, a pre-trained network is selected and adapted to the specific problem, even though the network was trained on another dataset (Quinn et al., 2018).

2.2. State of the art of applications

As state before, remote sensing-based machine learning applications can be found profusely linked to the urban context and in minor amount to displaced population settings. A brief overview of the two cases is given in the next sections.

As already introduced in Chapter 1, the feasibility of applications of remotely sensed data and imagery in these contexts is strongly dependent on the spatial, temporal, spectral resolution of the sensor (Quinn et al., 2018). From now on, it will be referred to resolution of imagery following the already stated ESA classification of resolution of sensor of their missions (ESA, 2015):

- VHR1 Very High Resolution 1 where resolution <=1m
- VHR2 Very High Resolution 2 where 1m < resolution <=4m
- HR1 High Resolution 1 where 4m <resolution<=10m
- HR2 High Resolution 2 where 10m <resolution<=30m
- MR1 Medium Resolution 1where 30m <resolution<=100m
- MR2 Medium Resolution 2 where 100m <resolution<=300m
- LR Low Resolution where resolution >=300m

It is clear how according to different resolution the scope and objective of machine learning applications may change. If with VHR imagery it could be possible to detect features of interest such as buildings or shelters, with HR imagery the detection of these features become more challenging while with MR and LR imagery mainly land cover applications are allowed (Quinn et al., 2018).

2.2.1. Applications to general human settlements

Identifying and mapping human settlements can be seen as a broader issue of identifying and mapping displaced populations. The first is fundamental for both technical and humanitarian purposes, as knowing where people are is the first step in planning a response. Anyhow, accurate and up to date data of human settlements are still scarce or quite heterogeneous from the data quality point of view. Consequently, the development of techniques to generate such datasets is crucial (Herfort et al., 2019).

Plenty of applications of remote sensing for human settlements identification already exist. In this section, some of them that are considered to be particularly relevant in this context are going to be resumed.

The identification of human settlements can be generally classified as a binary classification problem, in which the "non-human settlement" class is extremely heterogeneous, including a variety of different land covers and uses (Patel et al., 2014). Several studies have leveraged the new availability of remotely sensed data with high temporal and geographical coverage such as Landsat and Sentinel to implement models of human settlements detection. While some of them relied on cloud infrastructures such as Google Earth Engine (GEE) and the identification of relevant dataset to optimally identify human settlements (e.g. spectral indices, night time lights, etc.), others recurred to crowdsourcing in combination with deep learning techniques in order to optimize the mapping results.

Two studies conducted by Goldblatt et al. (Goldblatt et al., 2016; Goldblatt et al., 2018) focused on the binary supervised classification of urban land cover over very large geographic areas (country scale) using high-resolution satellite imagery from the Landsat 7 and 8 missions, which allows a resolution of 30 meters.

The first study aimed at building and providing an open-source labelled dataset of urban areas in India and at applying it for a supervised urban areas classification exercise. The second study is focused on identifying alternative datasets such as remotely sensed night-time lights data than can be associated to urbanization and can improve the detection of urban areas. Both the studies leveraged the Google Earth Engine cloud computing platform and Landsat data and both studies perform a pixel-based classification instead of an object-based classification, because of the higher computational power required by object-based classification and the large unit of analysis.

The first study relies only on spectral bands of the Landsat imagery as inputs to the supervised classifiers, with an addition of two indices: Normalized Difference Vegetation Index (NDVI) and NDBI (Normalized Difference Built-up Index). Three classifiers are tested: Classification and Regression Tree (CART), Support Vector Machine (SVM) and Random Forest. The second study, instead, combines

the use of DMSP-OLS (Operational Line-scan System of the Defense Meteorological Satellite Program) night-time lights data to automatically extract and collect training examples of urban areas, and uses these training data for urban areas classification with Landsat imagery. The second study relies on a uniform artificial hexagonal tessellation mapping approach. This approach is used to collect training data and build a classification model independently for each hexagonal mapping unit. This allows to account for regional variations of the land cover at large geographic scales, to better achieve the desired zone mapping approach. In the second study a Random Forest classifier is trained.

Both studies lead to consider Google Earth Engine as an optimal and accessible platform for image classification and analysis over large and heterogenous geographical areas, when appropriate ground-truth data are inputted. In addition, both studies underline how classification performance improves when either additional datasets are used as input of the training dataset construction or indices are added to the imagery spectral bands. The importance of identifying the optimal parameters to be inputted to the classifiers is highlighted. Classification performance improves as well the larger the size of the training dataset, especially of the non-urban samples. Finally, they showed that Random Forest classifier's performance is higher to other classifiers when applied at large geographic scale and with noisy and high dimensional data (Goldblatt et al., 2016; Goldblatt et al., 2018).

Two studies carried out instead in the context of OpenStreetMap (Herfort et al., 2019; Goldblatt et al., 2020) tried to combine the power of the crowd with secondary datasets in order to optimize the results of collaborative mapping efforts in the context of human settlements mapping.

The first study combined crowdsourcing with deep learning algorithms to obtain optimal mapping results. In the study by Herfort et al., a task allocation strategy is developed to allocate classification either to deep learning or to crowdsourcing based on a confidence score. This study highlighted how critical are the lack of datasets tailored to remote sensing applications and the vulnerability of such applications to several factors including sensors sensitivity to environmental conditions and imagery quality. The benefit of such application is the combination of the precise intervention of human validators and the high performance of automated methods, which is delineated as the optimal approach for efficient human settlements mapping (Herfort et al., 2019). The second study, instead, was intended to estimate the completeness of OpenStreetMap buildings over some areas of interest. The task was carried out by interpolating secondary open data, such as night-time lights (VIIRS) and Sentinel 1 and 2 data. These data demonstrated to correlate very well with building footprints coverage over the areas of interest. Indeed, the combined effect of these data explained between 92% and 94% of the OpenStreetMap building footprints variations. This study highlighted how fusing night-time and daytime remotely sensed information can increase the separability of urban and non-urban land cover types. The overall aim of this study was to identify gaps in OpenStreetMap coverage. Indeed, once a Random Forest

Regression model linking selected variables and OSM completeness was trained, it was possible to launch it over unknown areas and identify areas where the actual building coverage did not match the predicted one, hence gaps in OSM completeness and where mapping efforts should then be focused (Goldblatt et al., 2020).

2.2.2. Application to displacement contexts

Main existing applications of machine learning in remote sensing for displacement settings refer to a set of methods comprising pixel-based classification, object-based classification, and mathematical morphology, but the issue is still open and under discussion and research (Quinn et al., 2018).

An identified example of application of machine learning for displacement settings detection is the study by Quinn et al. in 2018 (Quinn et al., 2018). The designed exercise was a deep learning application aimed at training a Mark-RCNN model over very high-resolution satellite imagery for single structures extraction within refugee settlements. A Mask-RCNN (Region-proposal Convolutional Neural Network) model simultaneously predicts the bounding boxes of objects, the class of belonging of the objects and it also performs a pixel segmentation in order to obtain the extent of each object detected in the image (Quinn et al., 2018). These experiments of training of the Mask-RCNN model led to the development of Pulse Satellite tool, a human supported tool that is able to analyse satellite imagery assisted by a Mask-RCNN neural network. The human contribute is inputted as feedback loops until a satisfactory result is obtained by the network. This allows the tool to be fine-tuned according to the specific user cases (e.g. a different environmental condition compared to the one the model was trained at the beginning) (Logar et al., 2020).

The experiment of Quinn et al. showed that it is possible through deep learning to detect a large proportion of structures within the studied settlements. However, the large variation of both quality and type of imagery used and of the features to be detected themselves led to the necessity of an interactive learning approach, where human can provide inputs and adjustments to reach usable accuracy levels. Trivially, the best results were obtained when the settlements to be analysed showed similar characteristics with the training dataset and when the structures of the settlements are easily distinguishable (Quinn et al., 2018).

A very similar tool to Pulse Satellite illustrated above is the platform Picterra (https://picterra.ch/). Picterra is an extremely intuitive online platform allowing the users to train machine learning algorithms to detect and map objects on satellite and drone imagery without directly dealing with the algorithm itself. Picterra allows the user to upload the imagery, train and build a detector by providing training and testing areas and training features and to run the trained detector over every image chosen by the user and export the objects detected. Picterra has been used, among

others, for detection of informal settlements shelters in Tanzania from drone imagery (Picterra, 2020).

In 2020, Thinking Machines and iMMAP partnered to investigate the possibility to apply artificial intelligence on high-resolution satellite imagery to detect and located new migrant informal settlements in Colombia as a consequence of the Venezuelan crisis for humanitarian aid. The project aimed at accelerating the detection of newly formed informal settlements in all municipalities with high incidence of migrants and refugees from Venezuela between 2015 and 2020, in order to speed up the field validation process. A machine learning model was trained to identify difference between informal settlements and formal settlements or other type of land uses. The model was trained on ground-validated field data of informal migrant settlements collected by iMMAP and on a random sampling of all the other features. High-resolution Sentinel-2 satellite imagery available in Google Earth Engine were used both as yearly composites and as spectral indices as input for the model. The final output is a probability map of informal settlement presence that was twice validated both through independent visual inspection of very high-resolution satellite imagery based also on change detection between years and through field validation. At the end of the project, 350 potentially new settlements were identified by the model across Colombia, of which 70 were already validated by field campaign (Thinking Machine, 2020).

Figure 2.1 Figure 2.1 The probability map of informal settlements presence (Thinking Machine, 2020).

These few examples taken both from scientific literature and from operational experiences such as the Thinking Machines one, suggest that the application of remotely sensed imagery in the field of displacement is being explored and it is providing interesting outcomes. Nevertheless, at the moment, the context reveals that there are still few applications successful in obtaining such a level of robustness and accuracy to be actually deployed in a fast and consistent way in an actual crisis situation.

2.3. Spatio-temporal machine learning and deep learning

As briefly illustrated, there is an increased availability of big spatio-temporal datasets fastening the applications of fully or semi-automatic processing and analysis of these data. While classical machine learning exercise involving random forest or support vector machines algorithms have also been applied in remote sensing applications to manage temporal series of datasets, deep learning has been employed mainly on single temporal scene or imagery (Lubej et al., 2018). Machine learning methods in remote sensing rarely simultaneously describe both spatial context and temporal variations (Lubej et al., 2019).

The work carried out by Lubej et al. (Lubej et al., 2018) illustrates a deep learning application to leverage both spatial and temporal relationships in satellite imagery, in order to classify the land cover of Slovenia. To this end, annual series of Sentinel-2 images f 2017 were used. Briefly, the aim of the exercise was to use a Temporal Fully Convolutional Network (TFCN) able to leverage the time-series to maximise the classification results (Lubej et al., 2019).

2.4. The gap: application of high-resolution open data

After the evaluation of existing applications of machine learning methods in the displaced population fields, a gap was identified. Mostly all applications, except for the one carried out by Thinking Machines and iMMAP of which unfortunately no scientific paper has been published yet, were performed on very high-resolution satellite imagery (below 4 meters resolution). This cuts out a considerable amount of freely available satellite or alternatively remotely sensed products, mostly high-resolution products. These HR products could be investigated and exploited for developing semi or fully automated data management and analytical methods (Witmer, 2015), also for application in displacement settings.

What would be the benefits of recurring to open high-resolution remotely sensed products?

- First of all, the high temporal and spatial coverage of high-resolution missions. High-resolution satellite missions such as Sentinel-2 mission have a coverage between latitudes 56° south and 84° north and a revisit time of 10 days at the equator with one satellite, and 5 days with 2 satellites under cloud-free conditions which results in 2-3 days at mid-latitudes. Contrarily, very high-resolution missions are unable to store all the imagery they could potentially collect of the Earth because of the capacity of the system itself.

Consequently, they are operated on request or on a tasking basis. For some areas, such as remote areas or developing countries, the most recent very high-resolution imagery available could be several years old (Quinn et al., 2018). Historical analysis and strict monitoring are by far more feasible with high-resolution products than with very high-resolution products.

- Very high-resolution satellite missions, because of the stated storing capacity problem, operate on request and rotate their sensors in order to capture the requested area of interest. Consequently, the angle of incidence varies image by image. The angle of incidence varies also in images of the same area with different sensing time, even if taken by the same sensor. This leads to inconsistencies in calibration and in difficulties in comparing images. Contrarily, high-resolution missions have sensors capturing almost the entire Earth according to a predefined grid system and a predefined measurement process. This makes this type of imagery more prone to be used in machine learning exercises, because models can be transposed across space and time with less difficulties compared to very high-resolution imagery. The latter, indeed, while providing a higher resolution and consequently a wider range of applications, require to work around dataset shifts and model generalization issues (Quinn et al., 2018).

- Open high-resolution remotely sensed imagery is free of charge and plenty of platforms and infrastructure exist nowadays to explore, process and download these data. This extends the possibility to use this imagery to monitor changes in almost real-time.

Of course, as previously stated, the range of applications potentially supported by HR satellite imagery is narrower than VHR. Nevertheless, as it will be showed in Chapter 3, HR satellite imagery still allows the displaced population camps or settlements to be identified in their location and in their extent. Certainly, variation and limitation occur according to the type of feature and the environmental conditions. Despite that, the possibility to identify the presence, the location and potentially the extent of displacement contexts from HR imagery is considered from the author a potential benefit in:

- Providing a first idea of the displacement context with available data (Witmer, 2015).

- Identifying in the first stages of an emergency where to focus further mapping efforts, potentially with VHR satellite imagery.

- Performing high temporal resolution monitoring exercises with freely available and ingestible imagery, leveraging the newly released cloud infrastructures.

2.5. Limits and risks

It is of extreme importance to not forget themes such as privacy, sensitivity, and ethics (Quinn et al., 2018) when dealing with high and very high-resolution imagery covering potentially every corner of the world. This is especially valid when dealing with vulnerable population, even if the main objective of such applications is intentionally positive.

When such imagery become public, vulnerable population become automatically exposed. And exposal does not always turn into improvements of the condition of vulnerable population. One outstanding and negative example is the 2007 Amnesty International's project Eyes on Darfur. This project consisted in feeding regular updates of satellite imagery covering 12 vulnerable villages in Sudan's province of Darfur. The main ideally beneficial aim was to monitor any sign of violence and conflict committed to these villages and put pressure on the government and finally reducing the violence itself (Reuters, 2007). Main back motivation was the inability of organization to reach the field because of government's restriction, hence satellite imagery was a workaround information access. Unfortunately, this project received as much attentions as negative and impactful were its consequences. Indeed, a rise in violence was identified in the villages monitored by the project in the years following the project itself, leading the project to shut down in 2008. It is supposed that higher visibility and government's retaliation were main causes of this increased violence (Forbes, 2020).

Humanitarian contexts are highly challenging. Timing and context knowledge are fundamental in analysis the feasibility of a project and its costs and benefits, especially on the interested population. Furthermore, when dealing with ethics, it should be kept in mind that most of the time populations monitored by such projects have no knowledge at all of being monitored (Forbes, 2020).

Additionally, a high degree of accuracy is expected by the products of these kind of applications, in order not to worsen the situation providing approximative information to whom must actually intervene on the field. Remotely sensed imagery is affected by many variables and inaccuracies related to climate, time, sensor quality, etc. (Quinn et al., 2018). Humans' intervention can contribute to them as an additional variable and inaccuracy.

Chapter 3

3. Sentinel-2 data for displacement contexts

As introduced in Chapter 2, high-resolution imagery such as Sentinel-2 data exhibits a spectrum of benefits and potentialities that are considered to be beneficial in emergency contexts and specifically in emergencies related to displaced populations. The possibility to assess displacement settings with field campaigns are often time-constrained and limited by cost and security factors, despite being fundamental in assessing their size and evolution. Consequently, remote observations offered increasing support in providing evidence of displacement phenomena and in performing accurate mapping exercises, even though mostly through visual interpretation yet. Visual interpretation requires skilled analysts, and it is severely time and labour consuming, impacting and limiting its applicability over large scale emergencies that may require strict and regular temporal monitoring. This can be alternatively supported by a more automatic or semi-automatic approaches when their accuracy proves to be sufficient (Quinn et al., 2018). Nevertheless, the potentialities of high-resolution imagery (4 to 30 meters resolution) are still sparsely and rarely explored in the context of displacement.

This Chapter is going to present an overview of the Sentinel-2 mission and of its potential applications for detection, classification and monitoring of displacement settings. The following Chapter 3, instead, is going to provide an overview of potential indirect applications of the same data to assess indicators of ongoing displacement.

3.1.   The Sentinel-2 mission

Sentinel-2 is a wide-swath, high-resolution and multi-spectral imaging mission. Sentinel-2 is one of the European Space Agency next-generation Earth observation missions launched in June 2015, with the main objective of a high-resolution,

multispectral and high revisit frequency land monitoring. Land monitoring includes vegetation, soil and coastal areas but not exclusively, covering also humanitarian relief operations, disaster control and risk mapping. The Sentinel-2 mission is made of a constellation of two polar-orbiting satellites placed in the same sun-synchronous orbit and phased at 180 degrees to each other. Sentinel-2 mission as a revisit time at the equator of 10 days for each single satellite and 5 days if both satellites are considered, under cloud-free conditions. This is translated in a revisit time of 2-3 days at mid-latitudes due to overlap between swaths from adjacent orbits with different viewing conditions. Its swath width is of 290 kilometres and its coverage limits are from between latitudes 56º south and 84º north.

Sentinel-2 satellites are equipped with a single Multi-Spectral Instrument (MSI) optical sensor able to sample 13 spectral bands, four of which at 10 meters resolution, six of which at 20 meters resolution and three of which at 60 meters resolution. Sentinel-2 products available are Level-1C Top-Of-Atmosphere reflectance products and Level-2A Bottom-Of-Atmosphere reflectance products, for which the granules, also called tiles, are 100x100 km2 ortho-images in UTM/WGS84 projection. These tiles can be fully or partially covered by image data; indeed, the tiles grid does not correspond to the sensor swath so partially covered tiles correspond to those tiles at the edge of the swath (ESA, 2020).
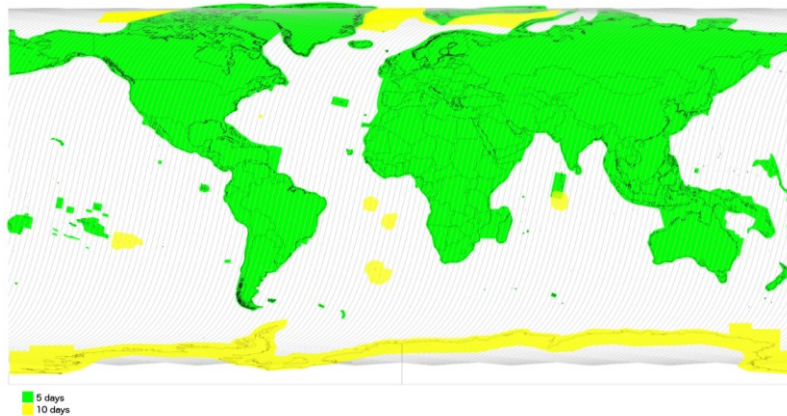
Figure 3.1 Complete observation scenario of the Copernicus Sentinel-2 constellation, with a revisit frequency of five days worldwide, since February 2018 (ESA, 2020).
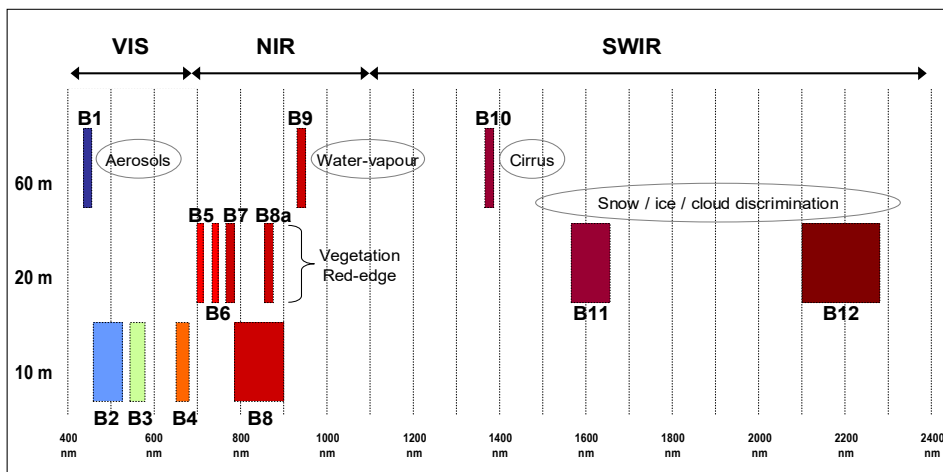
Figure 3.2 Spatial and spectral resolution of Sentinel-2 (ESA, 2020).

44

3.2. The construction of reference dataset

The preliminary step of a classification exercise is the construction of a reference dataset. A reference dataset is fundamental for training a classification algorithm in case of a supervised classification and for validating the classification algorithm in case of both a supervised and an unsupervised classification.

The selection of the dataset and of the number of data constituting the dataset itseld is an open issue in the machine learning field. As a rule of thumb, the larger is the training dataset the model is trained on and the better the model will be. As such, the larger the validation dataset and the higher confidence will have the results of the classification model.

A quite conspicuous number of labelled datasets have been developed in recent years for applications of remote sensing based supervised classification exercises. One of the most recognized source of commercial satellite imagery and labelled training data for the purpose to be used in machine learning research is the SpaceNet database (SpaceNet, 2020). Datasets exist for a variety of features (roads, building, etc.), even though dataset developed specifically for urban applications are still considered scarce (Goldblatt et al., 2016).

In the specific case of displacement settings (refugees' camps/settlements, IDP camps/settlements) there is no (to the author's knowledge) recognized and unified open ground-truth labelled dataset available that could possibly be used for a satellite imagery-based classification exercise. In order to implement classification exercises of displacement settings, a dataset of this kind of was built as first step.

3.2.1. Creation of the dataset

The workflow for the creation of the displacement settings dataset is resumed in the following figure.
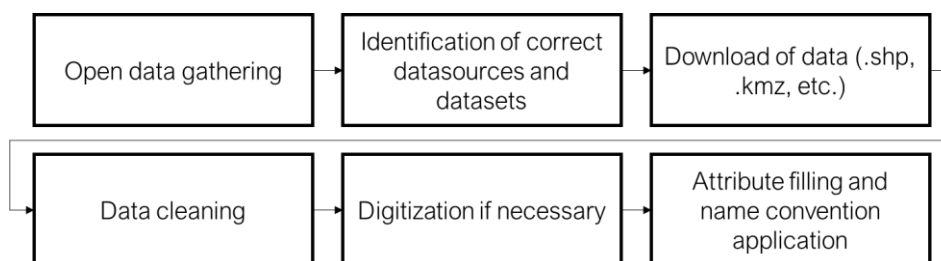
Figure 3.3 Workflow for the generation of the displacement settings dataset.

Sources of geospatial data and information regarding displacement settings are quite scattered. The openly available sources used in this process to collect data and georeferenced information and data on displacement settings were:

- The Copernicus Emergency Management Service (EMS) platform (https://emergency.copernicus.eu/mapping/list-of-components);
- The Humanitarian Data Exchange (HDX) platform (https://data.humdata.org/);
- The OpenStreetMap (OSM) data sharing apps (https://overpass-turbo.eu/);
- The Geonames platform (https://www.geonames.org/) coupled with Google Earth platform;
- The Humanitarian OpenStreetMap Team (HOT) Tasking Manager projects (https://tasks.hotosm.org/project/5668?task=85#bottom);
- The Displacement Tracking Matrix (DTM) database of the International Organization for Migration (IOM) https://displacement.iom.int/

Across these open sources, data were searched using key words such as #IDPcamp #refugeecamp #informalsettlement #displacedpeople. Most of data were retrieved in a *shapefile* format. Where no shapefile was available, it was possible to retrieve GPS coordinate of the displacement setting. Once data were retrieved from the differed sources, they had to be pre-processed. Main steps of pre-processing were:

- Cleaning of the data to keep only relevant features;
- Digitize the data when necessary (lack of proper shapefiles or noted changes over satellite imagery);
- Application of a naming convention to the shapefile. The naming convention chosen is XYZ_YYYYMMDD.shp where XYZ is the three-letter code of the country of belonging of the features and YYYYMMDD is the date of the source of the data, ideally the date of the imagery used to digitize the feature;
- Filling of the attributes of the data with relevant information, as by table below.

Attribute	Description	Values
src_date	Date of the source of the data (e.g. date of the imagery used to digitize)	YYYY-MM-DD
ext_date	Date of the digitization process (if known)	YYYY-MM-DD
source_nam	Name of the imagery used for the digitization process	
settl_type	Type of settlements (if known)	IDP_camp refugee_camp informal_settlement

		temporary permanent structured unstructured
other_info	Other relevant info on the displacement setting. Can be also a combination of the two e.g. temporary_structured	temporary permanent structured unstructured
event_type	Type of event that brought to displacement (if known)	hum nat
country	Three letter abbreviation of the country where the site falls into (see doc)	e.g. BGD
place_nam	Name of the place	
src_link	Link at the source of the data, important to retrieve further information if needed	
S2L2A	Is the site visible/available in S2L2A imagery?	yes no

Table 3.1 Attribute table applied to the displacement settings dataset.

A total of 117 displaced population settings data were collected. In addition to this, a dataset of 56 IDP camps was added only in Iraq. Indeed, for Iraq a very comprehensive census of IDP camps was released by IOM in August 2019. This census contained the GPS locations of the IDP camps in Iraq. Through Google Earth Engine platform these IDP camps were well digitized and added to the previous dataset. These camps were highly regular and similar across the country, being officially recognized camps.

The following map represents the geographical and numerical distribution of the collected dataset.
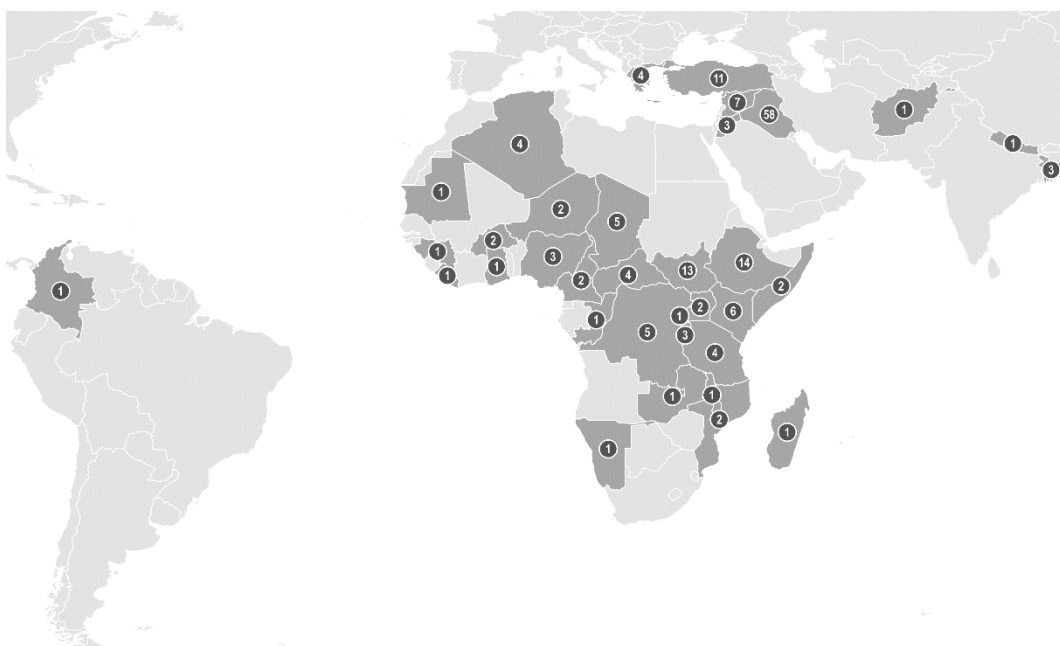
Figure 3.4 Geographic and numerical distribution of the displaced people camps/settlements dataset.

Update: as of April 2020, two websites are found to be under development sharing unified UNHCR and secondary sources data on GPs locations of populations of concern. These datasets can be potentially exploited to enrich the described dataset.

- https://unhcr.maps.arcgis.com/apps/webappviewer/index.html?id=2028db44801d43fe8eb49321eea19285
- https://maps.unhcr.org/en/apps/campmapping/index.html

3.3.   Examples and limitations

In this section, a list of charts and images illustrating the spectral response of samples of displacement settings, urban areas and comparative samples of vegetation and water over Sentinel-2 L2A data in four different countries are illustrated. The aim is to investigate and to highlight the expected challenges in dealing with separability between displacement and urban contexts, which often exhibit very similar spectral response, such as in the cases of Ethiopia and Uganda, or can be slightly separated from the spectral point of view, such as in the case of Iraq, when the settings are clearly made of different materials, size, and shape.
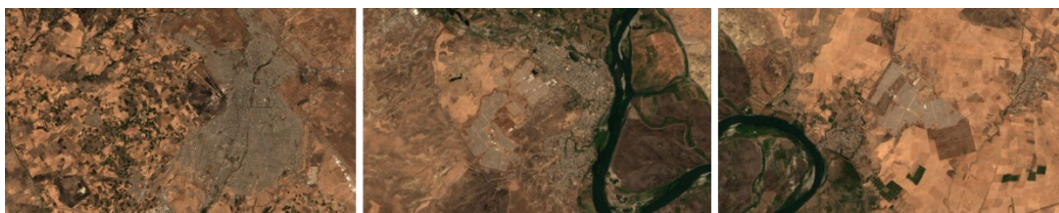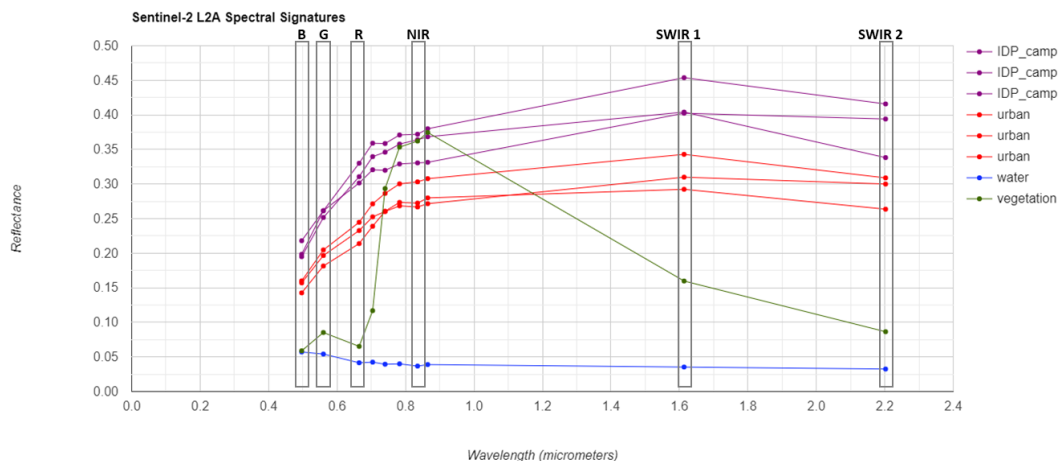
Figure 3.5 Spectral distribution of formal IDP camps (second and third image), urban areas (first image) and vegetation and water in Iraq over Sentinel-2 L2A image of August 2019.

48
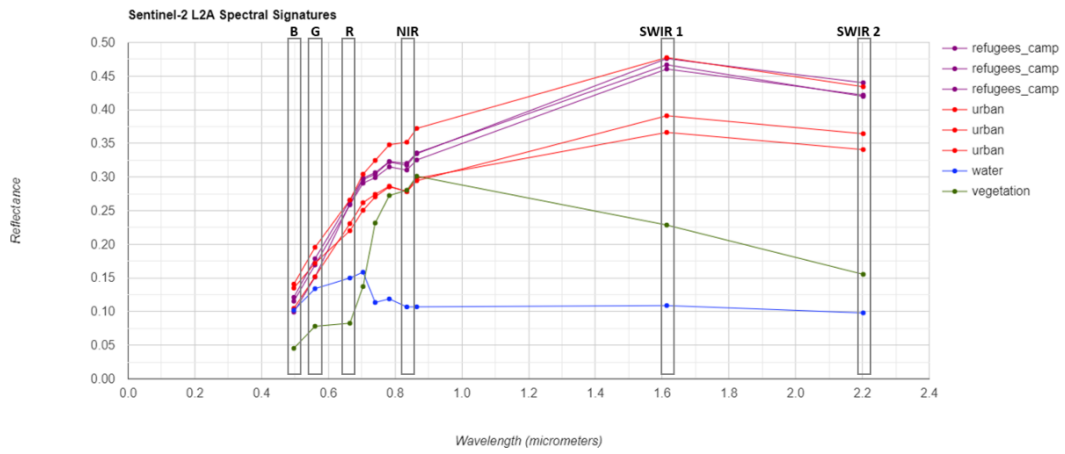
Figure 3.6 Spectral distribution of formal refugee camps (first and second image), urban areas (third image) and vegetation and water in Ethiopia over Sentinel-2 L2A image of September 2019.
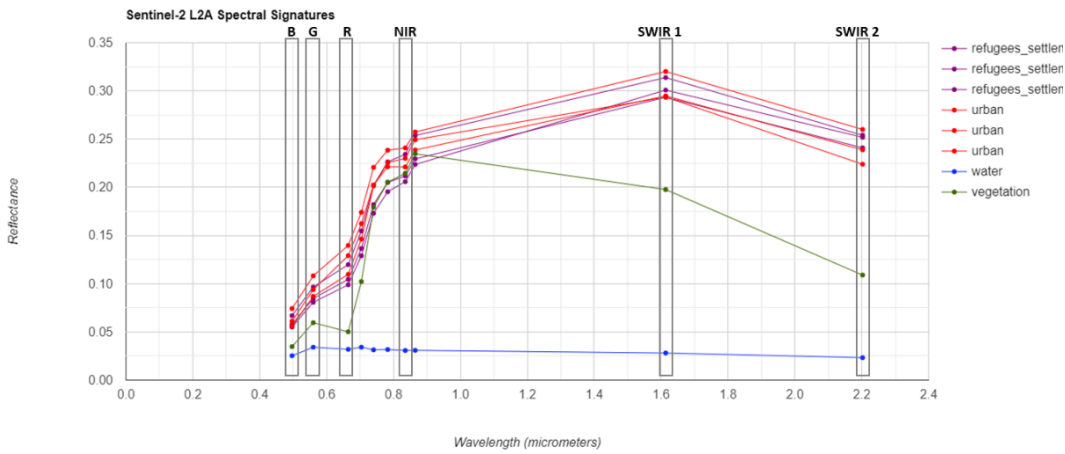
Figure 3.7 Spectral distribution of refugee settlements (first and second image), urban areas (third image) and vegetation and water in Uganda over Sentinel-2 L2A image of January 2020.
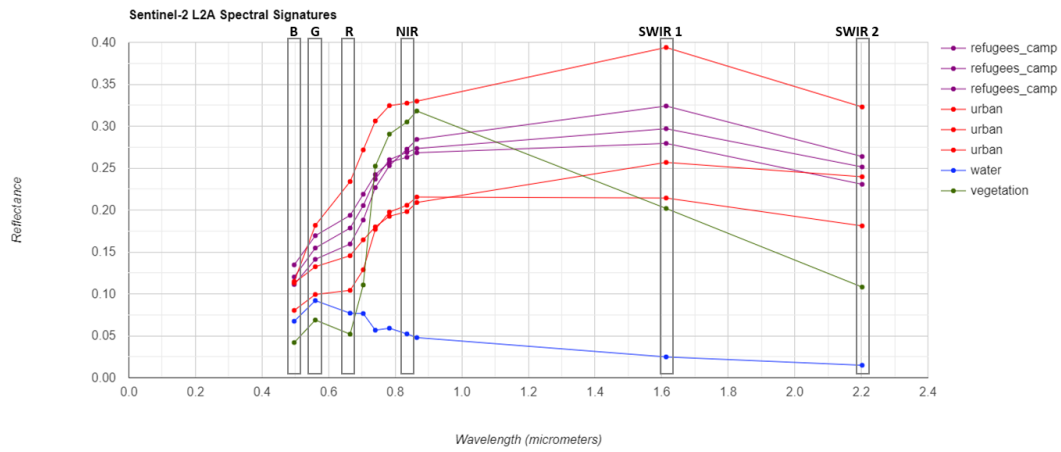
Figure 3.8 Spectral distribution of refugee camp (second image), urban areas (first and third image) and vegetation and water in Bangladesh over Sentinel-2 L2A image of March 2019.

The following charts, instead, represent the temporal evolution of monthly median Normalized Difference Vegetation Index (NDVI) averaged over the same samples used in the previous charts for the two illustrative cases of Iraq and Uganda. The analysis of spectral evolution in the time domain may additionally highlight seasons where these features are more separable compared to other.
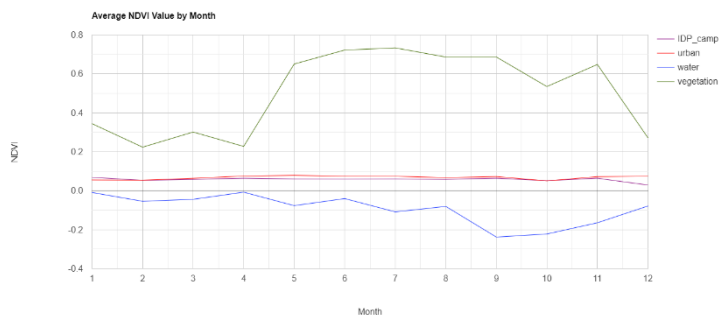
Figure 3.9 Temporal evolution of monthly median Normalized Difference Vegetation Index (NDVI) averaged over the same samples used in the previous chart for Iraq for year 2019.
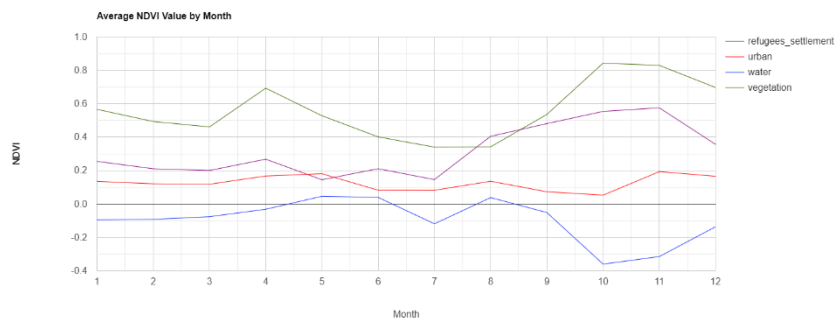
Figure 3.10 Temporal evolution of monthly median Normalized Difference Vegetation Index (NDVI) averaged over the same samples used in the previous chart for Uganda for year 2020.

3.4.   A binary classification problem

As already introduced, this Chapter will cover a series of classification exercises recurring to high-resolution satellite imagery and the collected dataset illustrated in the previous section. As stated in Chapter 2, the classification/identification of displacement settings can be defined as a binary classification problem, where two classes are identified:

- The "camp" class, defined as areas covered by structures ascribable to displaced population camps or settlements;
- The "non camp" class, defined as everything that is not included in the "camp" class.

The series of classification exercises were performed selecting different geographic areas, in order to obtain a spectrum of results involving different environmental, climatic and event-specific conditions. Indeed, as already discussed, the complexity and variety in types and distribution of displacement settings are one of the challenges in dealing with these features (Quinn et al., 2018).

In addition, an exploratory exercise was performed in the Picterra platform using the data collected about the IDP camps in Iraq through the IOM dataset. The aim of this exploratory exercise was to have a preliminary understanding of the potentiality of Sentinel-2 imagery in detecting and classifying displacement. The choice fell on the Iraq dataset in that it was related to ground truth data. The amount of data made it possible to have a fair amount of both training and testing data. In addition, these data showed to be very homogeneous in appearance and highly visible and detectable in high-resolution imagery, as shown in the image below.
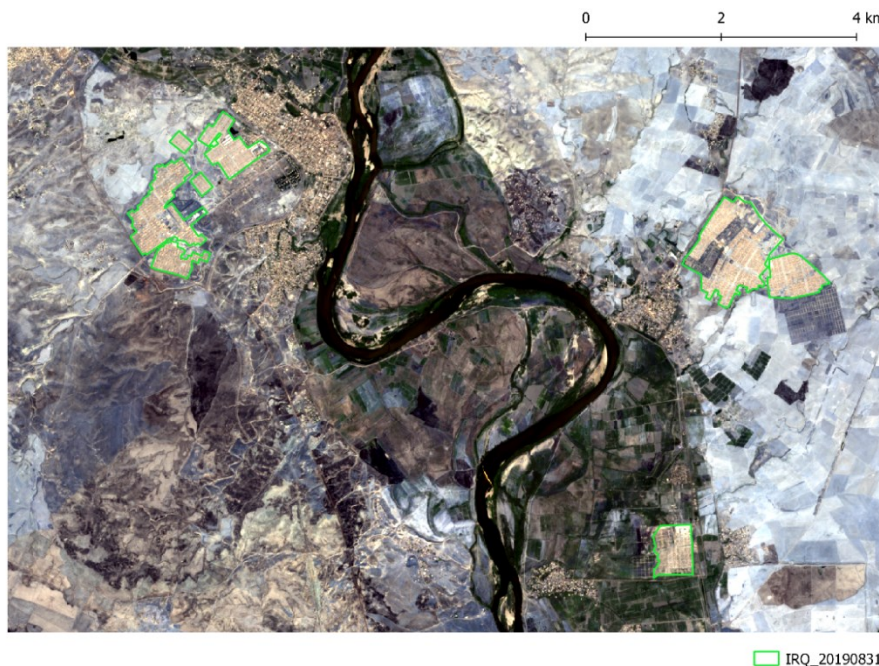
Figure 3.11 Few IDP camps in Iraq (in green) as digitized in the displacement dataset in an area near Mosul as seen in Sentinel-2.

3.4.1.   The Picterra platform

The aim of an exploratory classification was to understand how detectable are IDP camps features in high-resolution Sentinel-2 imagery from a qualitative point of view, recurring to a user-friendly platform such as Picterra, as a basis for further in-depth analysis recurring to other tools.

A full Sentinel-2 tile covering an area presenting a fair amount of IDP camps was selected at a date at which the dataset was collected, in order to have the features clearly visible in the imagery and aligned to the manually digitized dataset. The Sentinel-2 tile was selected and download from Copernicus Open Access Hub *S2A_MSIL2A_20190929T074711_N0213_R135_T38SLF_20190929T103511* (https://scihub.copernicus.eu/dhus/#/home). In the following image, the whole Sentinel-2 tile is presented and the IDP camps belonging to the collected digitized dataset from ground-truth data are also visualized in bright green. The area includes the city of Mosul and part of the city of Erbil.

Figure 3.12 Map showing Sentinel-2 image tested in this exercise and IDP camps belonging to the collected dataset (in green).

Over this whole tile, three sub-areas were identified and extracted in order to upload them in Picterra platform and to extract training and testing areas, they are illustrated in the following figure.

Figure 3.13 Map showing Sentinel-2 image tested in this exercise, IDP camps belonging to the collected dataset (in green) and sub-areas selected for the training and testing (in red).

For each of these sub-areas, a random selection of IDP camps included in these sub-areas was performed and these were used as training polygons in the platform. Over the overall Sentinel-2 tiles, 34 IDP camps areas (intended as polygons, hence more than one polygon may represent one single IDP camp) are present. It is decided to use 30% of these IDP camps areas as training data, hence 11 IDP camps areas were used as training data in the platform. A size limit was also set at the algorithm level, verifying the minimum size of the IDP camps available over the image and setting a minimum size of detection, in order not to detect smaller objects. Testing areas are used to iteratively check the results of the training at each run of the training and to adjust the identified training sample accordingly. This is a feedback loop with the user aimed at improving the training of the algorithm used within the platform before launching the classification over the whole image of interest. This procedure is performed for all three the chosen sub-areas. For each adjustment in training/testing areas the algorithm is re-trained. Once the iterative process of feedback loop between the platform and the user is concluded and the results shown over testing areas are satisfactory, the trained detector can be run over another image or over the entirety of the image.

The results of the detection were exported as points and imported in a GIS environment such as QGIS where a rapid and qualitative confrontation with the actual complete and ground-truth data of IDP camps could be performed. The interest was the detection of the object and not on the accuracy of the geometry detected. In the following image, the ground-truth IDP camps dataset is showed in green while the detection results from the algorithm trained in Picterra are shown as red points.
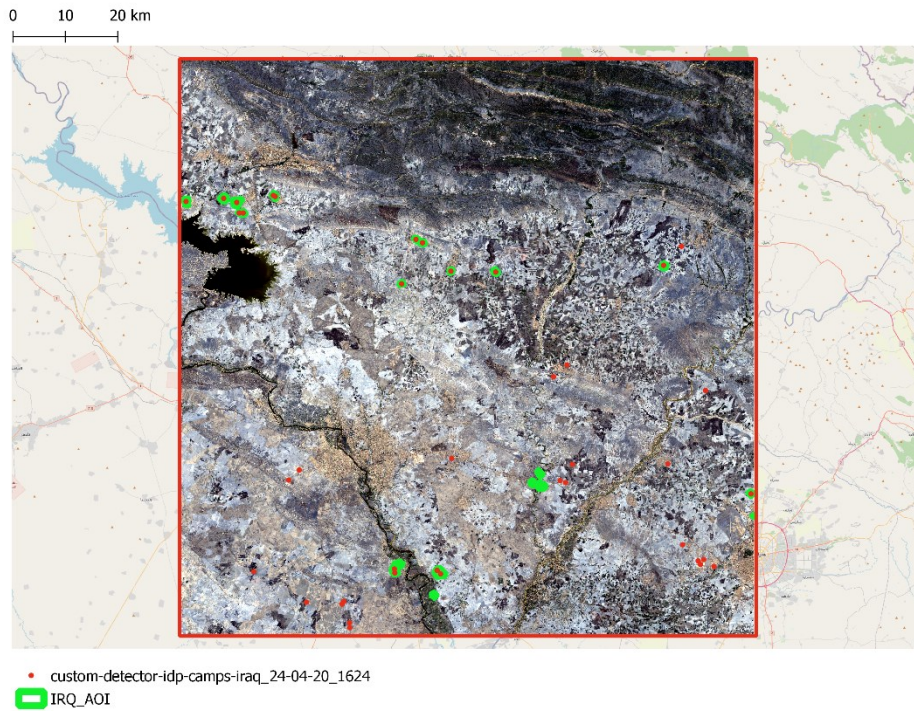
Figure 3.14 Map showing Sentinel-2 image tested in this exercise, IDP camps belonging to the collected dataset (in green) and IDP camps identified as such by the Picterra trained algorithm (red points).

As it can be qualitatively assessed from the image above, there is an evident presence of commission errors, also known as false positives, but also a few omission errors. Indeed, 65% of actual IDP camps (ground-truth) areas present in the image are intersected by areas identified as IDP camp by the platform (this percentage includes the 30% used as training data), this means that 35% of actual IDP camps in the image were not identified, so an omission error of 35%. In addition, only about 45% of detected objects intersects actual IDP camps areas, while the remaining about 55% represents commission error and so misclassified objects. In this preliminary phase of the analysis, a more comprehensive and rigorous error matrix was not calculated since the focus of the exploratory classification via Picterra was on the potential of the identification of the features and not on the geometric accuracy of their classification. A more quantitative approach will be carried out recurring to other tools in the next sections.

As expected, the majority of these misclassified objects were detected on features with spectral characteristics or shape easily attributable to the ones of IDP camps over the same image, such as agricultural fields or specific regularly shaped features such as industrial sites. These results suggest how the detection could be relevantly improved by adding more training examples belonging to the "non camp" class (such as agricultural fields, industrial sites, etc.), furthermore the result is slightly encouraging in that urban areas were rarely misclassified as IDP camps. Finally, even though the omission and commission error are still relevant, the result is encouraging in that it shows how displacement features could be potentially detected also in high-resolution open imagery such as Sentinel-2 imagery.

3.5. Unsupervised classification in Google Earth Engine

In order to assess the potential of high-resolution imagery such as Sentinel-2 in the detection and classification of displacement features, this section will explore a series of classification case studies. The case studies were selected in order to be quite heterogeneous in terms of geographical location, hence environmental conditions, and type of displacement itself.
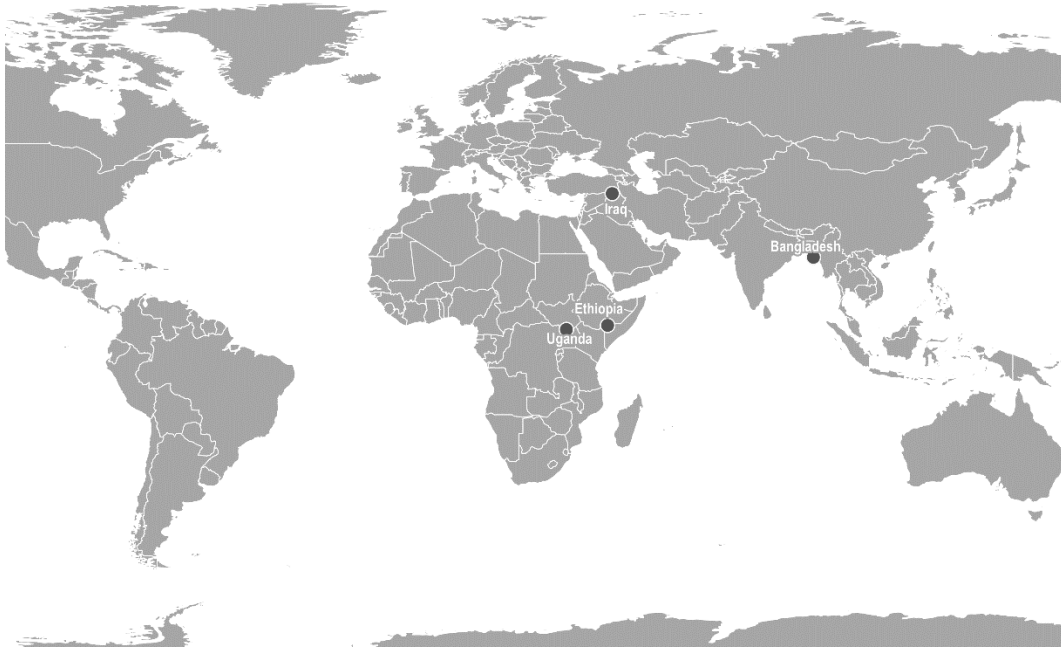
Figure 3.15 Location of the four case studies.

The Bangladesh case study concerns the large and well-known Kutupalong refugee camp near Cox's bazar linked to the Rohingya exodus from Myanmar. This camp showed a clear yearly evolution over the years, allowing a monitoring of its dynamics through the classification exercise. In Uganda, the Palorinya refugee settlement is considered as case study, which have all the characteristics of a long-term-village. The Ethiopia case study involves the two smaller refugee camps of Kobe and Melkadida, also considerable stable across years. Finally, in Iraq the focus is on the structured and well delimited Khanke, Kabarto 1, Kabarto 2 and Shariya Internally Displaced People's (IDP) camps.

The classification tests aim at assessing the accuracy of a pixel-based classification recurring to different sets of input classification features/properties, such as spectral bands, combined spectral indices or secondary datasets such as nocturnal lights.

3.5.1. Tools and methods

The exercises were entirely performed in the Google Earth Engine (GEE) platform (https://earthengine.google.com/) and the developed code is available in the Annexes.

The source data mainly used were:

- The displacement dataset manually digitized and already described and discussed in the previous sections;
- Sentinel-2 L1C Top-of-the-Atmosphere (TOA) reflectance data. This was a data-driven choice. TOA data were used instead of the L2A Bottom-of-the-Atmosphere (BOA) data because of their longer temporal availability on the GEE platform compared to the latter, available only starting from March 2017. Consequently, the Bangladesh case study could be investigated backwards up to 2016, and the reproducibility of the approach to other displacement contexts existing in years lacking availability of L2A products on the GEE platform is guaranteed. Rumora et al. study on the impact of different atmospheric correction methods (including no correction at all, hence using TOA products) on the accuracy of Sentinel-2 based land cover machine learning classification accuracy, concluded that the selection of the classifier overtakes in importance the selection of the atmospheric correction method, although their impact on the spectral values of the satellite imagery (Rumora et al., 2020). As already introduced, Sentinel-2 data are available at a resolution ranging from 10 to 60 meters according to the bands considered. Bands of Sentinel-2 data will be referred to as: B2 (Blue 10m), B3 (Green 10m), B4 (Red at 10m), B5 (Red Edge 1 at 20m), B6 (Red Edge 2 at 20m), B7 (Red Edge 3 at 20m), B8 (NIR at 10m), B8A (Red Edge 4 at 20m), B11 (SWIR 1 at 20m) and B12 (SWIR 2 at 20m);
- Monthly average radiance composite images using night-time data from the Visible Infrared Imaging Radiometer Suite (VIIRS) Day/Night Band (DNB), stray light corrected. VIIRS night-time data are composited monthly using a procedure to correct for stray light. In particular, the average DNB (Day/Night Bands) radiance values were used. The native resolution of this data is approximately 450 meters.

The overall methodology consists in the following steps:

1. Location of the displacement setting of interest and identification of the study site as the bounding box of the displacement setting of interest, potentially buffered if needed;
2. Image selection and image pre-processing;
3. Unsupervised pixel-based classification with k-means clustering recurring to different sets of input classification spectral

features/properties, with the objective to separate the "displacement setting" class from any other class;

4. Visual inspection of the resulting clusters and selection/labelling of the outcome cluster better representing the displacement setting feature;

5. Validation accuracy metrics of the identified clusters calculated using as reference data the ones coming from the manually digitized dataset of displacement settings;

6. Selection of the best performing set of classification spectral features/properties in terms of overall validation accuracy and representation of the results.

The image pre-processing consists in the generation of yearly per-pixel composites of all the available imagery with less than 5% cloud coverage over the reference year. Per-pixel composites take a collection of images available over the same geographical areas for a selected time window and reduce them per-pixel using different statistical aggregating functions resulting in a single image representative of the time window. It was decided to work with yearly per-pixel composites for mainly two reasons:

- To prevent or to avoid the impact of clouds in the final composites used for the representation and classification exercises. Working with optical imagery, a finer temporal compositing (e.g. monthly or weekly) would have led to cloud affected composites;

- Displacement settings with a lifetime (appearance, evolution, disappearance, or movement) shorter than a year exist, and the yearly composites make it difficult to properly capture or monitor such type of settings. Nevertheless, the selected study sites concern displacement situations that are observed to be quite stable even across several years and extensive. Indeed, the rapid evolution of a displacement situation is mostly typical of small, informal and emergency linked contexts, which are rarely observable from high-resolution imagery and hence not covered by this Chapter.

The image pre-processing phase also includes a data harmonization process. Since bands and imagery used to generate input classification data happen to have different spatial resolutions, they were all resampled to 10 meters resolution using a bilinear interpolation method and the pixels were aligned by reprojecting the data in the same coordinates reference system (CRS).

The reference year used for the generation of the per-pixel composites corresponds to the reference year of the manually digitized data used as input of the location of the study area and then as reference data for the validation accuracy metrics.

The image selection and pre-processing phase varies slightly according to the specific case study. In the case of Bangladesh, given the striking evolution of the displacement setting across the years, it was decided to perform a temporal monitoring of the setting itself. Consequently, five years from 2016 to 2020 were used as reference years for generating yearly composites and yearly classifications.

The following scheme resumes the applied methodology.
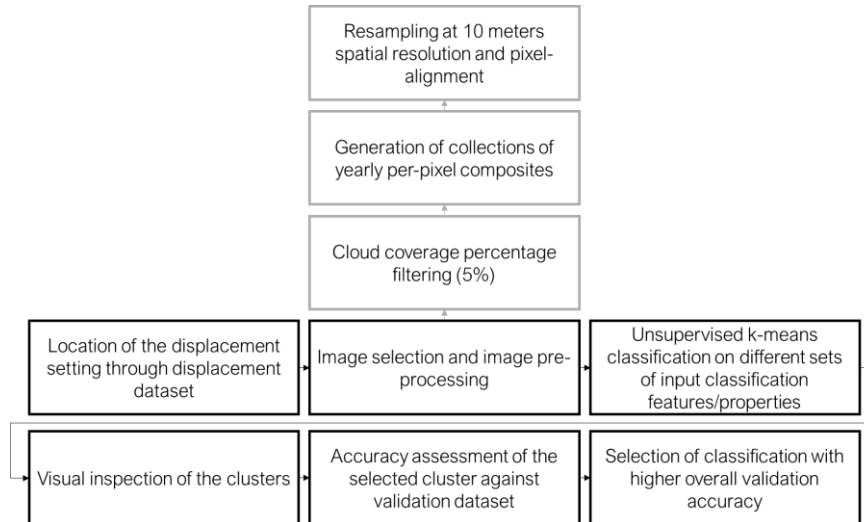
Figure 3.16 Methodology scheme.

3.5.2. Clustering

The selected approach for the classification exercises is the one of unsupervised classification or clustering, in that the main objective of the tests is to investigate the separability of displacement features from the surroundings in high-resolution imagery and the impact on the separability, hence on the classification accuracy, of different spectral features.

Again, the main objective here is trying to separate two classes: the "displacement setting" class as areas covered by structures ascribable to displaced population camps or settlements, and the "non displacement setting" class, defined as everything that is not included in the "displacement setting" class. To this end, the clustering algorithm outputs will be visually inspected until the algorithm identified clusters will align with the classes that are meant to be separated, even considering that the "non displacement setting" class will most likely be represented by more than one cluster that will be eventually merged into one.

Finally, the availability of the manually digitized and visually interpreted dataset of displacement features will serve as study sites identification and as reference for validation accuracy metrics calculation.

Clustering algorithms allows to partition numerical observations into clusters. The clustering algorithm generally requires to be instantiated with a set of parameters and to be trained using a user defined amount of training data. Once trained, the clustering algorithm can be applied on an image whose numeric

properties have to be clustered. Once the clusters are identified, they can be visually labelled, hence assigned with a specific class.

The k-means clustering algorithm was selected since it is the one clustering algorithm available in the GEE platform. The k-means clustering algorithm aims at partitioning observations into k clusters based on their features/properties, where k is user defined and each cluster is identified by its mean/centroid. The algorithm is iterative, starting with k clusters, assigning each observation to one of the clusters, calculating the mean/centroid of each cluster and reassigning each observation to the cluster with the closest mean/centroid. The mean/centroid are then recalculated for new clusters until convergence of the algorithm itself. Finally, each observation will belong to the cluster with the nearest mean of the user-defined feature/property. The main aim of the k-means algorithm is to minimize the intra-clusters variance. Additionally, the GEE platform provides the x-means clustering algorithm, including an efficient estimation of the number of clusters. In the following classification exercises, the optimal number of clusters was assessed via a combination of a preliminary use of the x-means clustering algorithm to identify the efficient estimated number of clusters and visual inspection of iterative results until a satisfactory classification outcome.

One benefit of clustering is that the same inputs should always reproduce the same outputs if the order of the inputs remains the same.

Within the GEE platform, the clustering algorithms implemented are from the Weka software. Their implementation may differ from the commonly clustering algorithms available in common remote sensing software, given the parallel distributed computing approach of Google Earth Engine. Indeed, common clustering algorithms build the model on the entire image intended to be clustered. The Weka GEE implementation of the clustering algorithms, instead, requires building ("training") the clustering model on a representative subset of the data, generally randomly selected, before being applied to the whole image to predict the clusters. This allows as well to train the clusterer on the totality of the pixels constituting the image that has to be clustered, matching common remote sensing software behaviour, but since the Weka classifiers implemented in GEE are not fast neither efficient nor robust, when training with as few as 10 bands * 100k points, it may potentially lead to memory errors in the platform (Google Earth Engine, 2021). Consequently, in the following classification exercises, k-means will be trained on a reduced randomly selected number of pixels of the image to identify the clusters and then applied to assign the remaining pixels to the identified clusters.

3.5.3. What features/properties can improve classification?

As stated before, the second aim after exploring the potential of high-resolution imagery in classifying displacement is to identify the best set of spectral features to this end. When performing an image pixel-based classification, different features/properties (one or many) of the image itself can be used as input parameters

to train and run an algorithm in the case of a supervised classification or to clusters into several classes the properties of the image to be classified in the case of an unsupervised classification. To this end, image stacks can be generated containing several bands/properties, where each one could be used as an input feature/property for a classification algorithm.

In conclusion, the aim of this Chapter is also to explore different combinations of features/properties that can be used when classifying displacement and their impact on the classification result and accuracy.

Spectral bands

The first inputs that can be used when classifying a satellite image are its spectral bands itself, hence the spectral surface reflectance at different wavelengths that constitute the different bands of a satellite image. In this case, since using per-pixel composites, the value and the date along the year of the spectral reflectance per pixel will depend on the statistical aggregating function used for compositing.

Indices

Spectral surface reflectance at different wavelengths, hence different spectral bands of a same satellite image, can be combined into spectral indices. Spectral indices are able to enhance the contribution of a specific land feature's (vegetation, water, burned area, urban area, etc.) property compared to the others. Spectral indices, indeed, generally leverage inverse relationships between two (or more) bands in which the specific land feature manifest a different behaviour. Consequently, spectral indices lend themselves to a variety of applications, from multi-temporal monitoring of vegetation health conditions to thresholding or other kinds of classification.

Two indices were taken into accounts in these classification exercises. The first is the widely recognized Normalized Difference Vegetation Index (NDVI). The NDVI is calculated as:

$$NDVI = \frac{(NIR - Red)}{(NIR + Red)}$$

Where Red and NIR are the spectral reflectance measurements acquired within the red visible and the near-infrared wavelengths regions, respectively. NDVI ranges from -1 to 1 and it leverages the high reflectance of vegetation in the near-infrared region and the strong absorption in the visible region to enhance vegetation with a high NDVI value.

The second index considered is the Normalized Difference Built-Up Index (NDBI). The NDBI is calculated as:

$$NDBI = \frac{(SWIR - NIR)}{(SWIR + NIR)}$$

Where SWIR and NIR are the spectral reflectance measurements acquired within the shortwave-infrared and the near-infrared wavelengths regions, respectively, with a SWIR band between 1.55-1.75 μm and a NIR band between 0.76-0.9 μm. NDBI ranges from -1 to 1 and it leverages the higher reflectance of urban areas in the SWIR region compared to the NIR region to highlight built-up areas.

Composites

The compositing technique allows to combine spatially overlapping images, such as a series of multi-temporal images over the same geographical area, into a single image where each pixel of the image is the result of an aggregation process or a specific function. The result of a composite implies that pixels at different locations in the resulting composite come from different times. This is valid for each band constituting the input images in the input collection. The compositing discussed here differs from the visualization composite that allows visualizing multi-bands data on a screen in real or false colours.
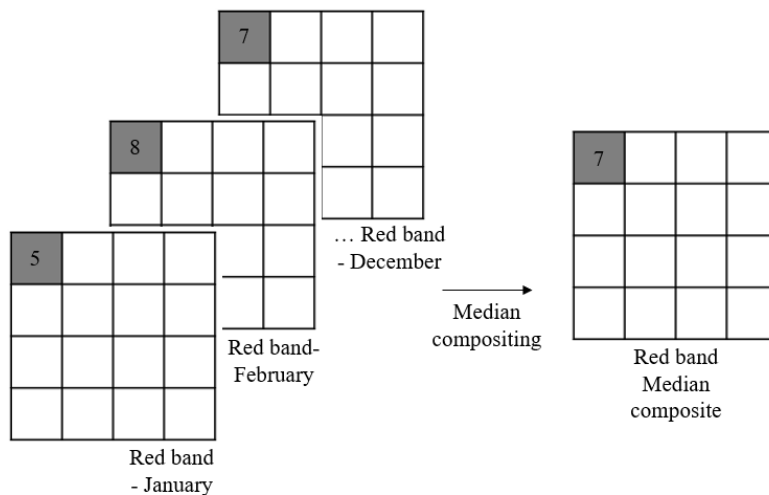
Figure 3.17 Compositing example.

One could composite a collection of images over the same geographical areas using different statistical aggregating functions. For instance, in the following classification exercises, one of the aggregating functions used will be the median of the pixel values for each band over a yearly collection of images. The median is well acknowledged because it returns a real pixel value to the composite. In addition, it reduces the impact of the presence of cloudy pixels within the collection itself, even if it was already filtered by cloud coverage less than 5%.

Composites can also be generated in order to maximize an arbitrary band in the input data, meaning that the resulting composite is created by picking each pixel of each band from the image in the collection that shows a maximum value at that pixel for the specifically selected bands. These types of composites are referred to as Quality Mosaics in the Google Earth Engine platform. One of these quality mosaics maximizes the greenness by selecting each pixel where the NDVI shows its maximum value. This type of composite can be referred to as greenest

61

composite, in that every pixel of the image will be picked at the time where the pixel itself expressed its maximum NDVI value hence it is supposed to have reached its peak in vegetation coverage or vegetation health condition. In the following classification exercises, the spectral features/properties obtained recurring to this specific type of composite will be referred to as "greenest".

Night-time light

The use of complementary and secondary datasets can also be taken into consideration when performing classification exercises of a specific phenomena. As fire detection products have been reportedly exploited to detect potential violence outbursts, night-time lights measures from satellite imagery have also been explored in humanitarian applications contexts. As an example, patterns of night-time lights have been used to monitoring unfolding conflicts as well as to correlate night-time light loss or fluctuation with the presence of IDP or large-scale populations movements (Quinn et al., 2018).

In this context, VIIRS Stray Light Corrected Nighttime Day/Night Band Composites Version 1 data are explored firstly to identify whether they provide additional information in the selected displacement case studies. In addition, they are exploited as an additional spectral feature to assess their impact on classification accuracies. VIIRS Stray Light Corrected Nighttime Day/Night Band Composites Version 1 data represent monthly average radiance composite images using night-time data from the Visible Infrared Imaging Radiometer Suite (VIIRS) Day/Night Band (DNB) at approximately 450 meters resolution available from 2014 up to now. These data are composited monthly, implying than over some regions the impact of clouds over a month might not be avoided. The fact that during the classification exercises they were composited into yearly median overcomes this issue These specific data have been corrected for stray light effect.

3.5.4.   Accuracy assessment

The output of a classification requires the assessment of its thematic accuracy as a quantitative indicator of the quality of the classification itself. The thematic accuracy of a classification is the agreement between the classification and a reference data that is taken into account as ground-truth. The result is a confusion matrix used to calculate the various accuracy metrics and whose diagonal elements represent the number of features correctly classified as the ground-truth class. This implies that a reference dataset to be used as ground-truth is needed in order to quantify the accuracy of a classification. In this case the visually inspected/manually digitized dataset of displacement setting features described in the previous section will be used as reference dataset.

While the diagonal elements of a confusion matrix represent the number of features correctly classified, three other accuracy metrics are worth being described:

- The overall accuracy is the ratio between the sum of all correctly classified features/pixels and the total number of reference features/pixels in the considered area.

- The producer's accuracy (PA) represents the accuracy from the point of view of the producer of the map, indicating how often real features on the ground will be correctly represented in the classification map. The producer's accuracy is calculated as the number of features/pixels in the reference correctly classified for a given class divided by the total amount of pixels in the reference for that class.

- The user's accuracy (UA) represents the accuracy from the point of view of the user of the map, indicating how often a class on the classification map will be actually present on the ground. The user's accuracy is calculated as the number of pixels correctly classified for a given class divided by the total amount of pixels classified as that class.

From the accuracy metrics it is also possible to derive the classification errors:

- The omission error is a measure of how many reference features/pixels were omitted from the classification (false negatives) and it is the complement of the PA.

- The commission error is a measure of misclassified features/pixels in a given class (false positives) and it is the complement of the UA.

3.5.5. Bangladesh case study

Context

The stateless Muslim minority of Rohingya started to flee an outburst of violence in Myanmar in August 2017. More than 742'000 Rohingya refugees sought shelter in nearby Bangladesh. The exodus started in 2017 and continued throughout 2018 (UNHCR, 2019b).

Most of the fleeing population settled in and around the already existing formal refugee camps of Kutupalong and Nayapara in Bangladesh's Cox's Bazar district. Kutupalong formal refugee camp already hosted 13'901 refugees before the 2017 exodus. The new influx of refugees led to the fast growth of spontaneous settlements and the onset of makeshift camps/settlements around Kutupalong itself, which eventually ended up in a continuous area including Kutupalong and Balukhali extension sites and unofficial camps in the more southern part (Reuther Graphics, 2017; UNHCR, 2019b).

Most of the makeshift camps were and still are in precarious sanitary, security and environmental conditions. Indeed, the Cox's Bazar district is highly affected by monsoon rains between May and September, posing severe risk to the volatile

infrastructures of the refugees' settlements, since most of refugees reside over terrain highly prone to landslides and flooding (UNHCR, 2019b).

Figure 3.18 Aerial view of Kutupalong refugee camp (TheInterpreter, 2018).

Kutupalong refugee camp is now one of the largest of its kind and hosts about 600'000 people in about 13 square kilometres (Reuther Graphics, 2017; UNHCR, 2019b). Kutupalong and the more southern Jamtoli-Bagghona-Hakimpara sites represent the current study sites.

Figure 3.19 Kutupalong and nearby makeshifts settlements/camps on 2017 May 26, three months before the influx of refugees, showing the green, hilly areas around the camps still relatively unoccupied (Reuther Graphics, 2017).

Study site and input data

The study site is illustrated in the following figure. The image is an RGB composite of the 2019 yearly median of a Sentinel-2 L1C. The green boundary represents the refugee camp outline manually digitized in the generation of the displacement dataset.

Figure 3.20 Study site.

A k-means clustering algorithm was trained on 100'000 randomly selected input pixels (15% of the totality of pixels of the classified image) with the following combinations of features/properties used as input of a k-means clustering algorithm:

- Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS;
- Yearly median of NDVI, NDBI;
- Yearly median of NDVI, NDBI, VIIRS;
- Yearly median of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12;
- Yearly median of NDVI, NDBI, B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12, VIIRS;
- Yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B2, B3, B4, B5, B11, B12 and yearly median of VIIRS.

Results

The optimal number of clusters was assessed to be $k = 4$ and once the cluster better describing the refugee camp outline was visually identified, it was possible

to calculate the validation accuracies metrics for the 2019 classification against the manually digitized data with reference year 2019. The best overall validation accuracy was achieved with the combination of yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS, compared to the other combinations of features/properties as illustrated in the Annexes. Indeed, the following graphs illustrating the spectral signature averaged in the whole camp area and the whole non camp area, suggests that higher separability occurs in these bands.

Figure 3.21 Spectral distribution of refugee camp and non-camp areas.

This selected combination of input features provided the following accuracy metrics expressed as error matrices, while the accuracy metrics for the remaining combinations of input features are illustrated in the Annexes.

VALIDATION ACCURACY

			Predicted			
Bangladesh		**2019**				
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS						
		Non-camp	Camp			
	Non-camp	471390	12381	483771		
Actual	Camp	52911	135296	188207		
	Overall	524301	147677	671978		
		Omission	*Comission*	*PA*	*UA*	
	Non-camp	0.025592687	0.100917221	0.974407313	0.899082779	
	Camp	0.281131945	0.083838377	0.718868055	0.916161623	
	Overall	0.902836105				

Figure 3.22 Combination of bands with the highest overall validation accuracy metrics for 2019 classification of the Bangladesh case study.

In the following figure, the result of the 2019 classification achieving the best overall validation accuracy compared to the manually digitized data is illustrated, highlighting where commission or omission errors occur and where agreement between the two data occur.

It has to be taken into account that a feature such as a refugee camp does not have clear outlines in most cases, unless in the case of a formal structured camp/settlement. As a consequence, even manually digitizing the outline can be challenging and susceptible to the subjective interpretation of the interpreter.

Figure 3.23 2019 refugee camp classification accuracy metrics.

In the next pages, instead, the whole temporal evolution of the data is illustrated from 2016 (left) to 2020 (right) of:

- RGB composites of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data;
- RGB composites of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data;
- Yearly medians of VIIRS night-time data.

In the second pages the outcomes of the clustering classification from 2016 (left) to 2020 (right) are illustrated together with a binary representation of the clusters, meaning that all clusters not representing the refugee camp feature have been dissolved into a single cluster and the remaining one is the cluster better representing the refugee camp feature. Again, all classification images refer to the classification achieving the best overall validation accuracy.

Despite it was not possible to evaluate validation accuracy metrics for all years given to the lack of a reference data, it is possible to visually assess the agreement of the classification with the current expansion of the refugee camp itself.
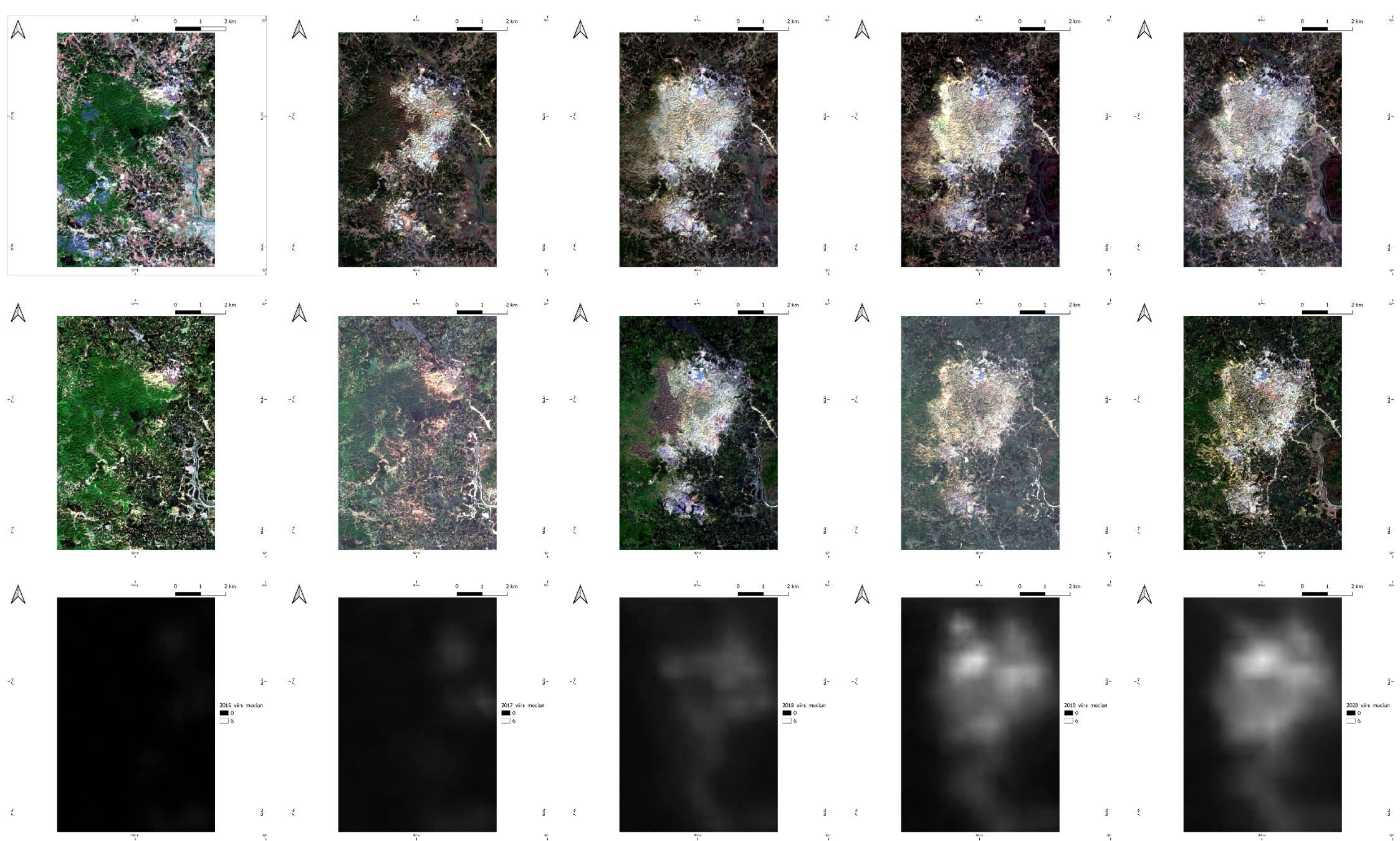
Figure 3.24 From the top: RGB composites of yearly medians composites of R, G, B bands of Sentinel-2 L1C data, RGB composites of greenest composites of R, G, B bands of Sentinel-2 L1C data, yearly medians of VIIRS night-time data.

Figure 3.25 From the top: yearly clusters resulting from k-means classification, the same clusters aggregated in order to highlight refugee camp and non-camp features.

3.5.6.   Uganda case study

Context

Uganda is the African country hosting the largest number of refugees due to its refugee's policy that used to provide, among other benefits, a cultivable piece of land (WFP, 2021). Most of incoming refugees flee the neighbouring countries of South-Sudan, Democratic Republic of Congo, and Burundi. Uganda is nowadays hosting more than 1'400'000 refugees, who gradually settled in the existing 30 refugee settlements existing across the country. About half of the totality of refugees' population in Uganda resides in the settlements of Bidi Bidi, Pagirinya and Rhino in the northwest part of Uganda (UNHCR, 2021a). Uganda refugee settlements have developed with time to assume all features of villages.

Figure 3.26 Bidi Bidi refugee settlement (The New York Times, 2018).

Palorinya refugee settlement is located in the North Wester region of Uganda in the Moyo district. This settlement was opened in December 2016 in order to lighten the pressure due to new arrivals at the Bidi Bidi refugee settlements. Most of refugees hosted at Palorinya settlements come from South Sudan. More than 42'000 refugees settled in Palorinya two months after tits opening. Palorinya refugee settlements is divided into three zones (01, 02, 03).

Study site and input data

The study site is illustrated in the following figure. The image is an RGB composite of the 2017 yearly median of a Sentinel-2 L1C. The green boundary represents the refugee settlements outline collected in the generation of the displacement dataset.

Figure 3.27 Study site.

A k-means clustering algorithm was trained on 100'000 randomly selected input pixels (2% of the totality of pixels of the classified image) with the following combinations of features/properties used as input of a k-means clustering algorithm:

- Yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B4, B7, B11, B12;
- Yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B4, B7, B11, B12 and yearly median of VIIRS;
- Yearly greenest of NDVI and yearly median NDBI;
- Yearly median of NDVI, NDBI, B4, B7, B11, B12, VIIRS;
- Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12;
- Yearly greenest of B4, B7, B11, B12.

Results

The optimal number of clusters was assessed to be *k* = 8 and once the cluster better describing the refugee settlements outline was visually identified, it was possible to calculate the validation accuracies metrics for the 2017 classification against the manually digitized data with reference year 2017. The best overall validation accuracy was achieved with the combination of yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12, compared to the other combinations of features/properties as illustrated in the Annexes. The following graphs illustrates the spectral signature averaged in the whole camp area and the whole non camp area. The difference in accuracies metrics considering exclusively the combination of yearly greenest B4, B7, B11, B12 was minimal.

Figure 3.28 Spectral distribution of refugee camp and non-camp areas.

This selected combination of input features provided the following accuracy metrics expressed as error matrices, while the accuracy metrics for the remaining combinations of input features are illustrated in the Annexes.

VALIDATION ACCURACY

Uganda	**2017**				
Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12					
		Predicted			
		Non-camp	Camp		
	Non-camp	4003503	82177	4085680	
Actual	Camp	309546	128627	438173	
	Overall	4313049	210804	4523853	
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.020113421	0.071769646	0.979886579	0.928230354
	Camp	0.706446997	0.389826569	0.293553003	0.610173431
	Overall	0.913409432			

Figure 3.29 Combination of bands with the highest overall validation accuracy metrics for 2019 classification of the Uganda case study.

In the following figure, the result of the 2017 classification achieving the best overall validation accuracy compared to the manually digitized data is illustrated, highlighting where commission or omission errors occur and where agreement between the two data occur.

Again, it has to be taken into account that a feature such as a refugee settlement does not have clear outlines in most cases, unless in the case of a formal structured camp/settlement. As a consequence, even manually digitizing the outline can be challenging and susceptible to the subjective interpretation of the interpreter.
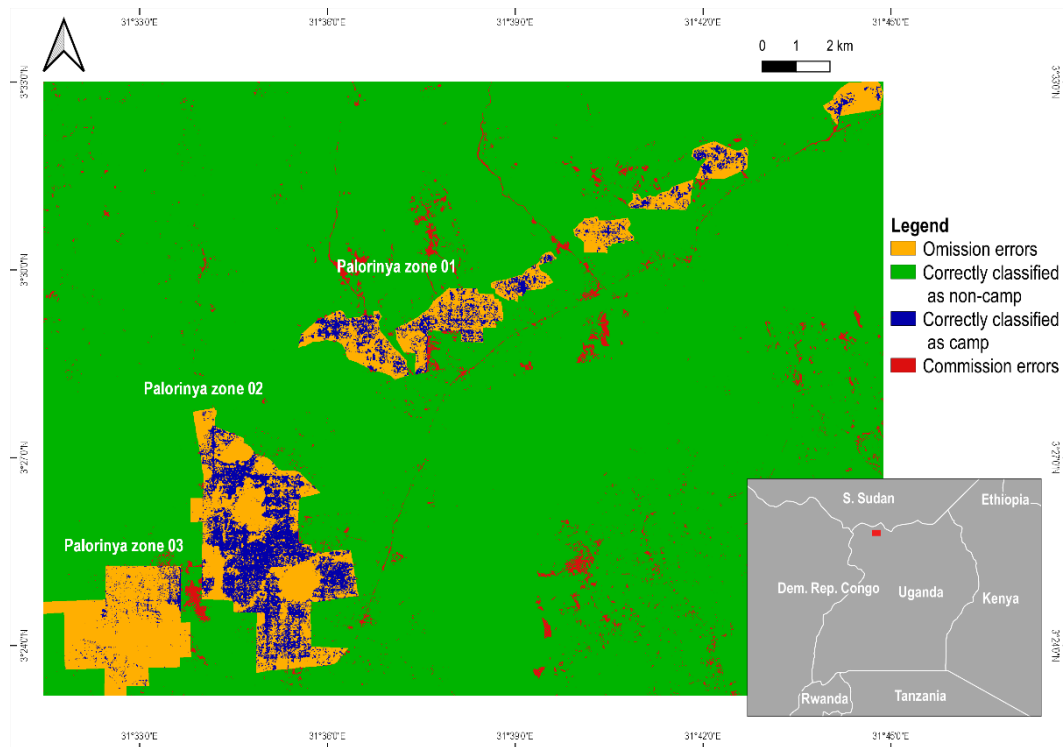
Figure 3.30 2017 refugee settlements classification accuracy metrics.

In the next pages, instead:

- RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2017;
- RGB composite of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2017;
- Binary representation of the clusters, meaning that all clusters not representing the refugee settlement feature have been dissolved into a single cluster and the remaining one is the cluster better representing the refugee settlement feature, for reference year 2017.

Again, classification images refer to the classification achieving the best overall validation accuracy.
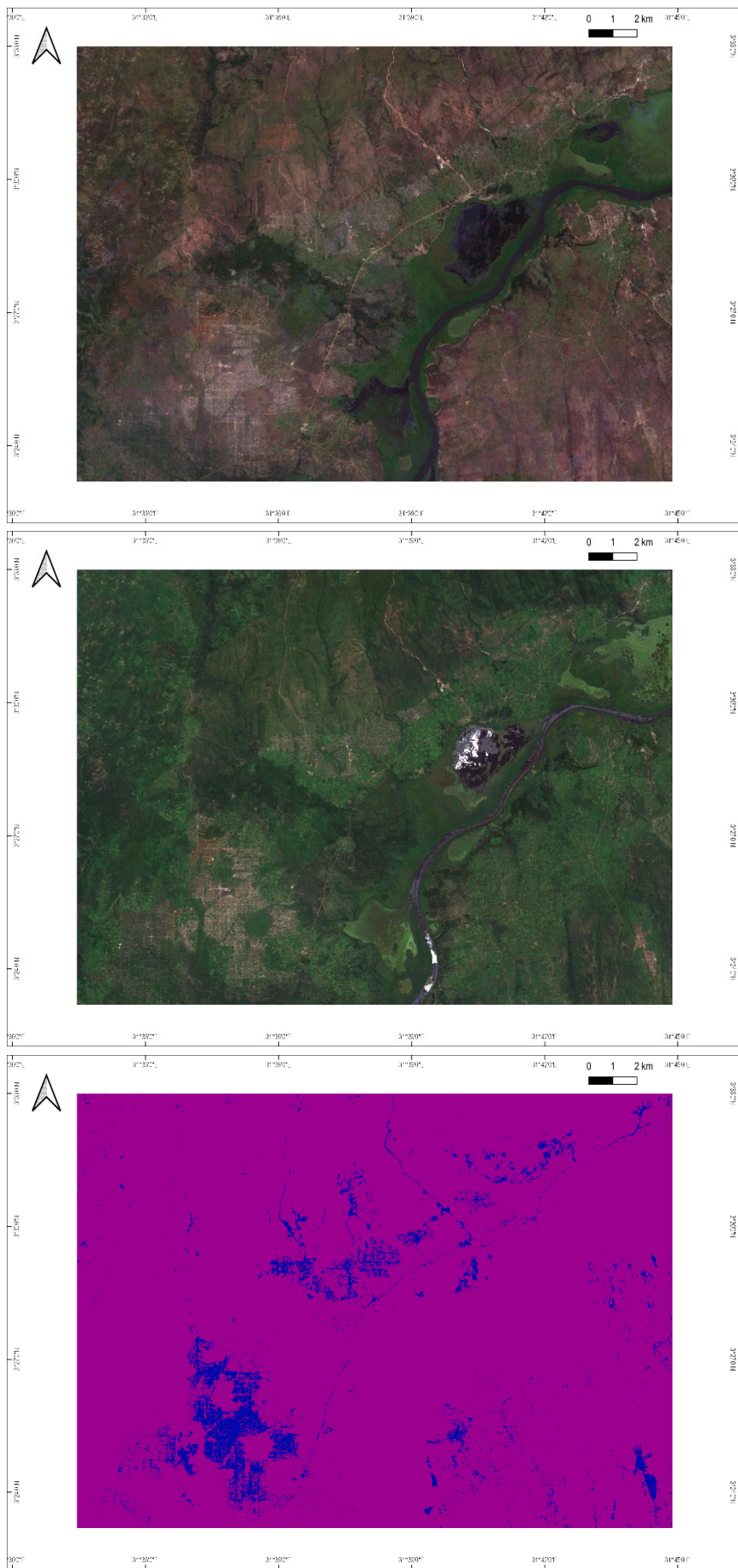
Figure 3.31 From the top: RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data, RGB composite of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data, clusters aggregated in order highlight refugee settlement and non-settlement features.

74

3.5.7.   Ethiopia case study

Context

Ethiopia hosted a total of 883'546 refugees at the end of September 2017, mostly accommodated in 26 refugee camps distributed across the country, which largely depend on humanitarian assistance. Ethiopia's long history of refugees' hospitality from mostly neighbouring countries of South Sudan, Somalia, Eritrea and Sudan is strongly linked to its open-door asylum policy (UNHCR, 2018a).

Eight refugee camps are located in the Somali region of Ethiopia, hosting mostly Somali refugees (Reliefweb, 2020). Among them, Kobe and Melkadida refugee camps were established in 2011 and they host now 31'127 and 35'038 refugees respectively (UNHCR, 2021).

Figure 3.32 Kobe refugee camp (Seed, 2015).

Figure 3.33 Melkadida refugee camp (Mapio, 2021).

Study site and input data

The study site is illustrated in the following figure. The image is an RGB composite of the 2019 yearly median of a Sentinel-2 L1C. The green boundary represents the refugee camps outline manually digitized in the generation of the displacement dataset.
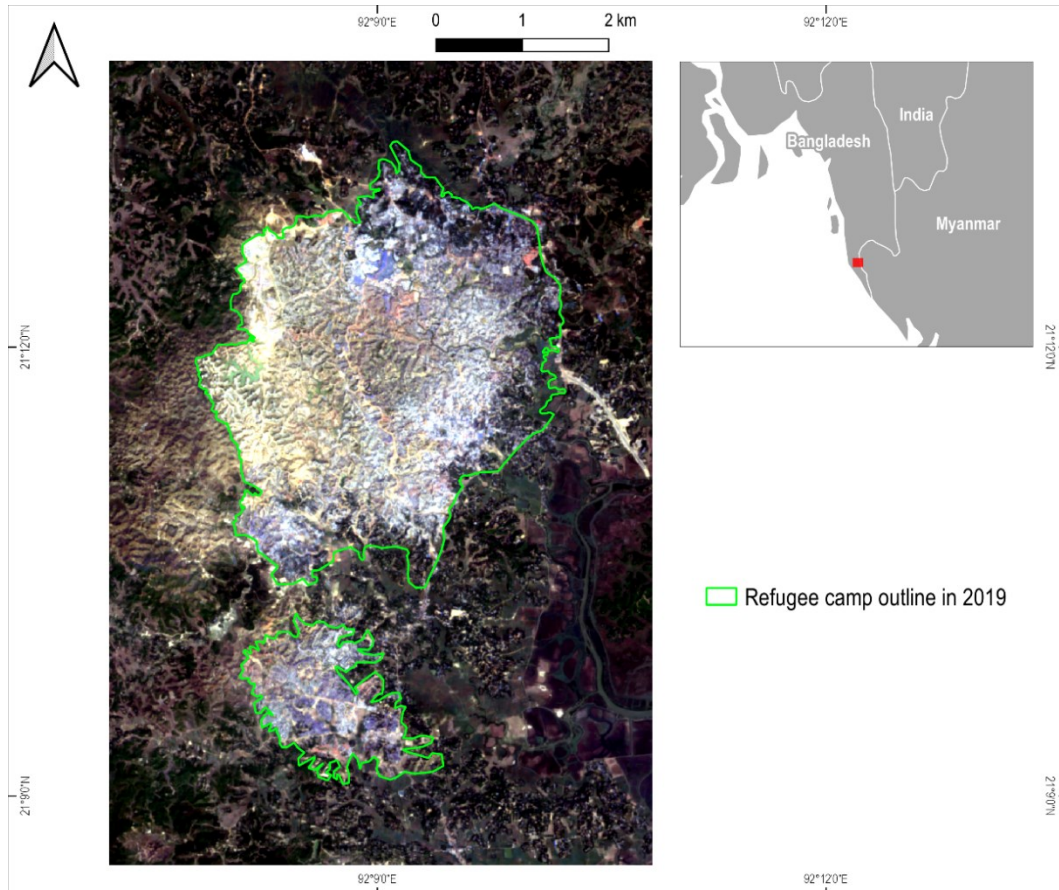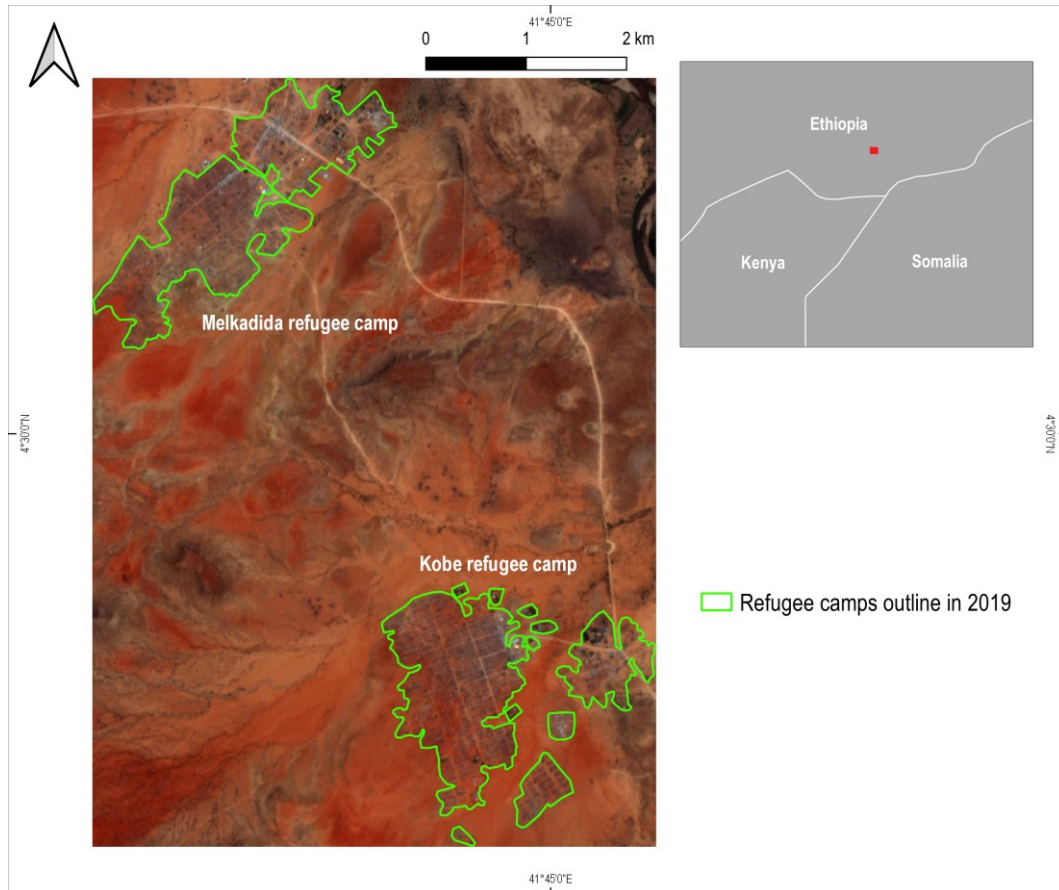

Figure 3.34 Study site.

A k-means clustering algorithm was trained on 100'000 randomly selected input pixels (23% of the totality of pixels of the classified image) with the following combinations of features/properties used as input of a k-means clustering algorithm:

- Yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B8A, B11, B12 and yearly median of VIIRS;
- Yearly greenest of NDVI, NDBI, B8A, B11, B12 and yearly median of VIIRS
- Yearly greenest of NDVI and yearly median of NDBI;
- Yearly greenest of NDVI, yearly median of NDBI and yearly median of VIIRS;
- Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12;
- Yearly median of NDVI, NDBI, B8A, B11, B12, VIIRS.

Results

The optimal number of clusters was assessed to be *k* = 7 and once the cluster better describing the refugee camps outline was visually identified, it was possible to calculate the validation accuracies metrics for the 2019 classification against the manually digitized data with reference year 2019. The best overall validation accuracy was achieved with the combination of yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B8A, B11, B12 and yearly median of VIIRS, compared to the other combinations of features/properties as illustrated in the Annexes. Indeed, the following graphs illustrating the spectral signature averaged in the whole camp area and the whole non camp area, suggests that higher separability occurs in these bands.
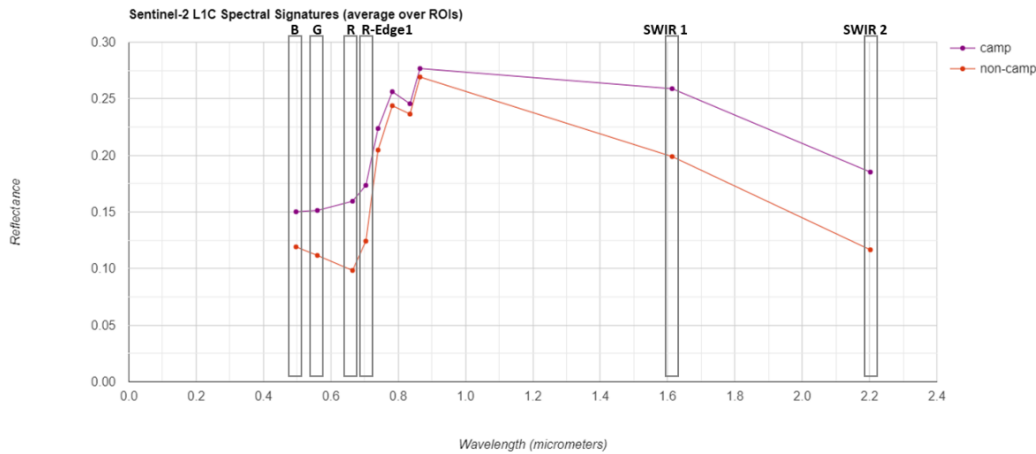

Figure 3.35 Spectral distribution of refugee camp and non-camp areas.

This selected combination of input features provided the following accuracy metrics expressed as error matrices, while the accuracy metrics for the remaining combinations of input features are illustrated in the Annexes.

VALIDATION ACCURACY

			Predicted			
Ethiopia		**2019**				
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B8A, B11, B12, VIIRS						
			Predicted			
		Non-camp	Camp			
	Non-camp	358854	12603	371457		
Actual	Camp	26760	32837	59597		
	Overall	385614	45440	431054		
		Omission	*Comission*	*PA*	*UA*	
	Non-camp	0.033928557	0.069395821	0.966071443	0.930604179	
	Camp	0.44901589	0.277354754	0.55098411	0.722645246	
	Overall	0.908681975				

Figure 3.36 Combination of bands with the highest overall validation accuracy metrics for 2019 classification of the Ethiopia case study.

In the following figure, the result of the 2019 classification achieving the best overall validation accuracy compared to the manually digitized data is illustrated, highlighting where commission or omission errors occur and where agreement between the two data occur.

Again, it has to be taken into account that a feature such as a refugee camp does not have clear outlines in most cases, unless in the case of a formal structured camp/settlement. As a consequence, even manually digitizing the outline can be challenging and susceptible to the subjective interpretation of the interpreter.
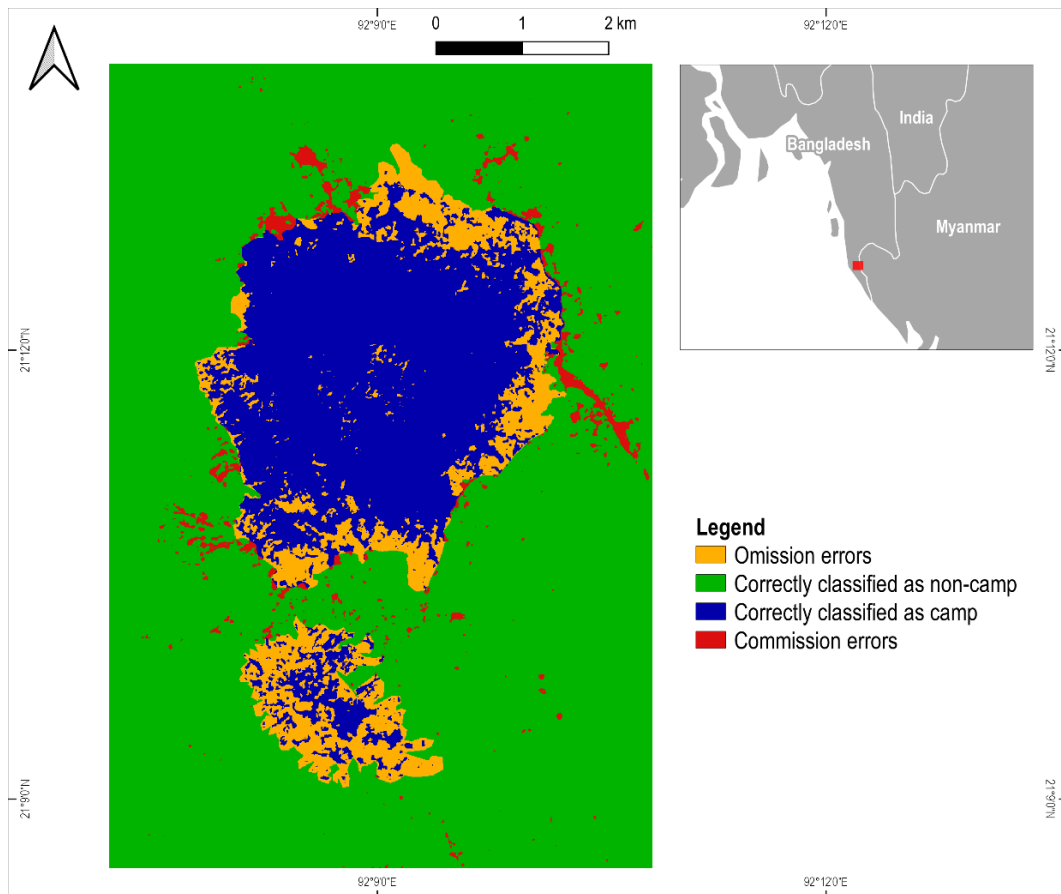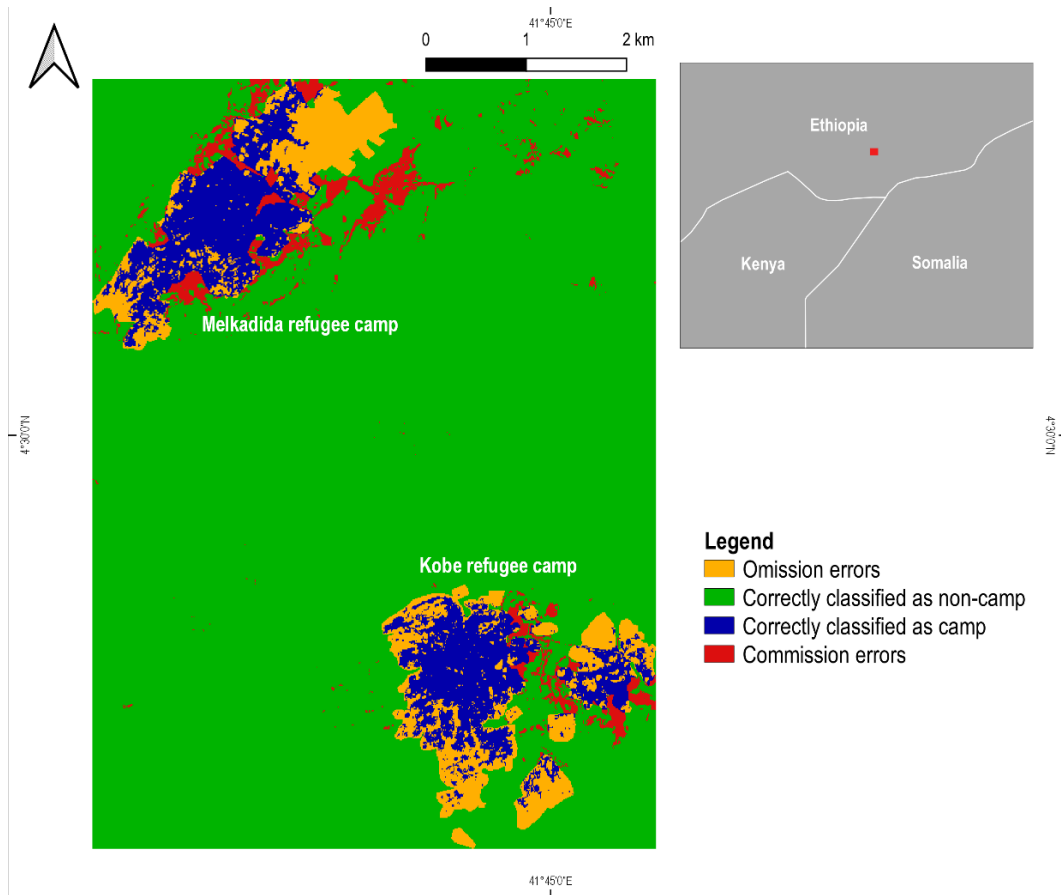

Figure 3.37 2019 refugee camps classification accuracy metrics.

In the next pages, instead:

- RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2019;
- RGB composite of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2019;
- Yearly median of VIIRS night-time data for reference year 2019;
- Outcome of the clustering classification for reference year 2019;
- Binary representation of the clusters, meaning that all clusters not representing the refugee camp feature have been dissolved into a single cluster and the remaining one is the cluster better representing the refugee camp feature, for reference year 2019.

Again, all classification images refer to the classification achieving the best overall validation accuracy.

Figure 3.38 Top from left to right: RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data, RGB composite of greenest composites of Red, Green and Blue bands of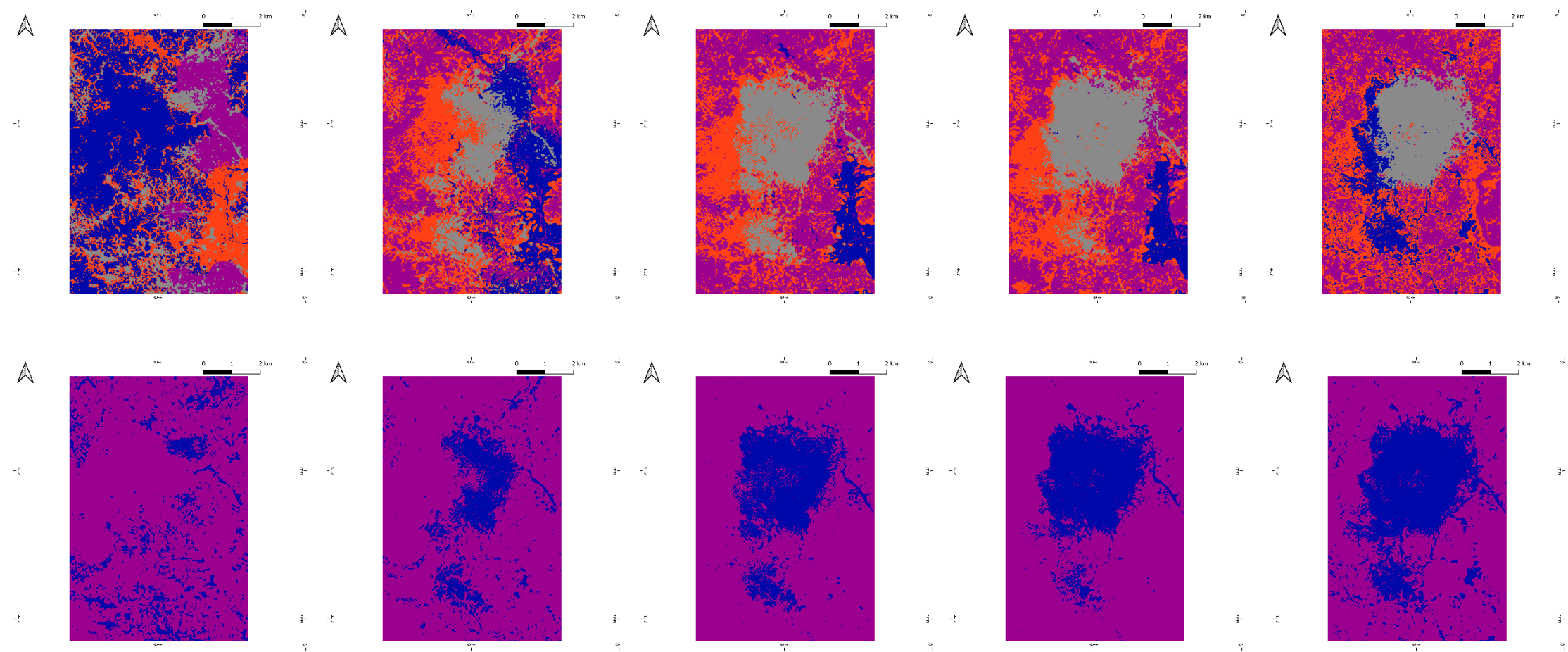 Sentinel-2 L1C data, yearly median of VIIRS night-time data. Bottom from left to right: clusters resulting from k-means classification, the same clusters aggregated in order to highlight refugee camp and non-camp features.

3.5.8.   Iraq case study

Context

Following the rise of the Islamic State of Iraq and Syria and its expansion in Northern and Central Iraq in 2014, a significant internal displacement of population occurred since civilians started to flee towards more safe and secure parts of the country (USAID, 2020). In February 2020 approximately 1.4 million Internally Displaced People (IDP) were enumerated throughout the country, of which mostly residing in formal IDP camps in Iraq (Reliefweb, 2020a), mostly managed by UNHCR (UNHCR, 2021b). The government decision to initiate a plan to close IDP camps to facilitate returns in 2019 has exposed this population to further instability (Reliefweb, 2020a).

The four formal IDP camps of Khanke, Kabarto 1, Kabarto 2 and Shariya, all opened in 2014 (Reliefweb, 2015), are located in the Sumel district north of the city of Mosul. They respectively hosted 14'192, 11'880, 12'073 and 15'270 IDP individuals by December 2020 (UNHCR, 2021b).

Figure 3.39 Khanke IDP camp (UNHCR, 2021c).

Figure 3.40 Kabarto IDP camp (Themenbuero, 2021).

Study site and input data

The study site is illustrated in the following figure. The image is an RGB composite of the 2019 yearly median of a Sentinel-2 L1C. The green boundary represents the IDP camps outline manually digitized in the generation of the displacement dataset.
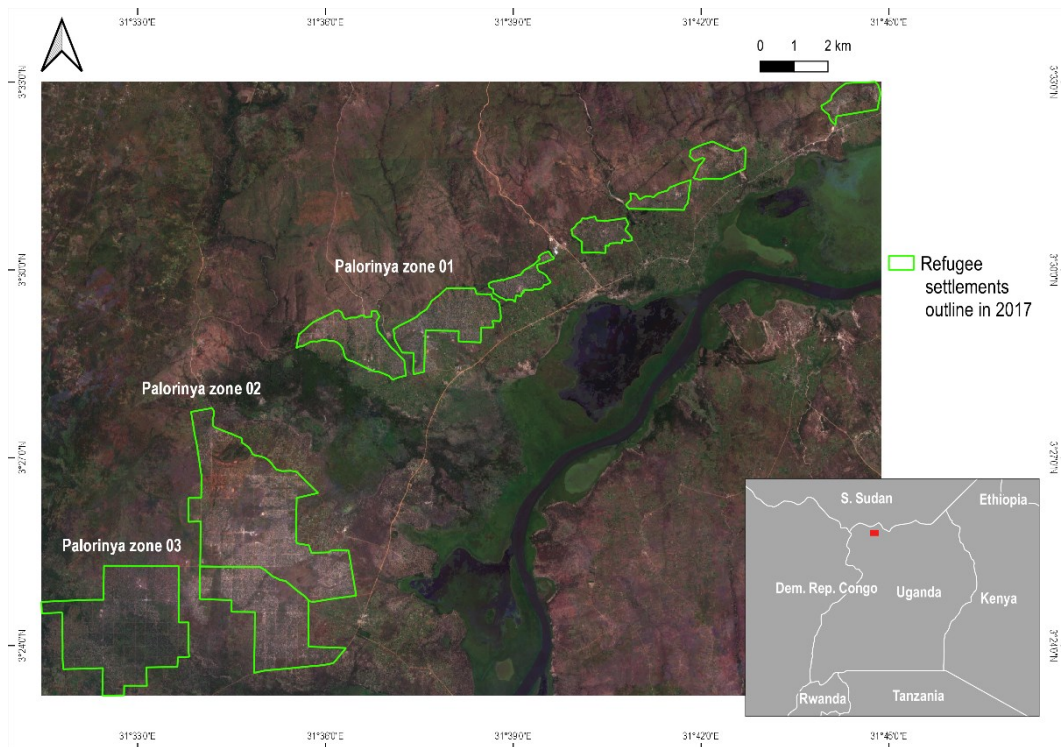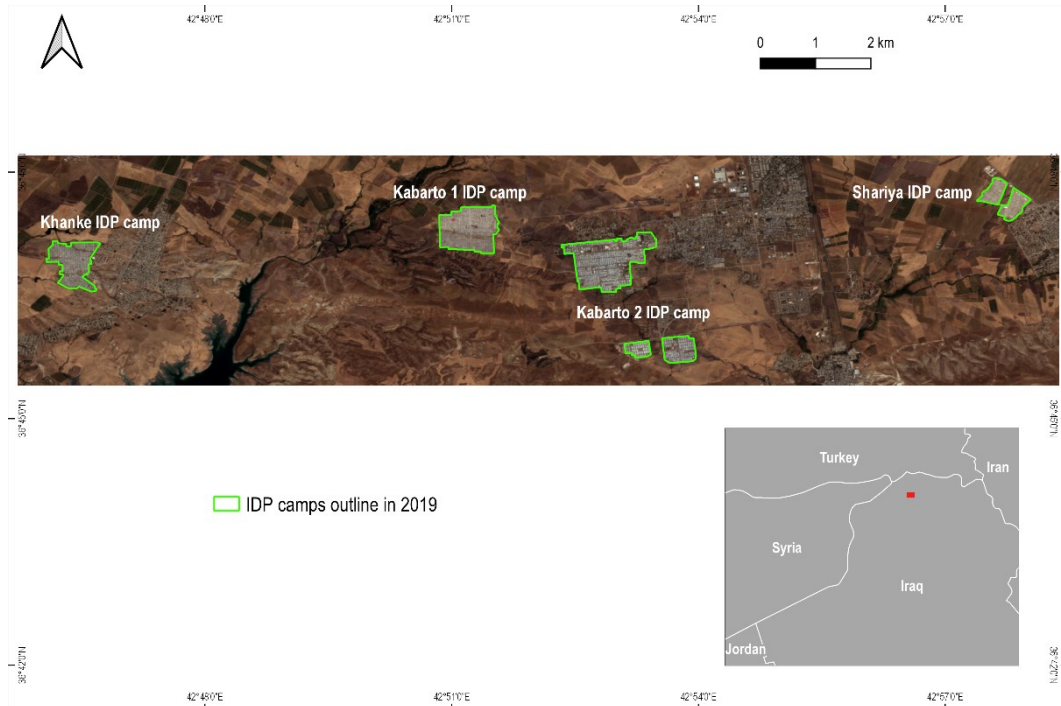

Figure 3.41 Study site.

A k-means clustering algorithm was trained on 100'000 randomly selected input pixels (8% of the totality of pixels of the classified image) with the following combinations of features/properties used as input of a k-means clustering algorithm:

- Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS;
- Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12;
- Yearly greenest of NDVI, yearly median of NDBI, yearly greenest of B2, B3, B4, B5, B11, B12 and yearly median of VIIRS;
- Yearly median of NDVI, NDBI;
- Yearly median of NDVI, NDBI, VIIRS;
- Yearly median of B2, B3, B4, B5, B11, B12.

Results

The optimal number of clusters was assessed to be $k = 8$ and once the cluster better describing the IDP camps outline was visually identified, it was possible to calculate the validation accuracies metrics for the 2019 classification against the manually digitized data with reference year 2019. The best overall validation accuracy was achieved with the combination of yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS, compared to the other combinations of features/properties as illustrated in the Annexes. Indeed, the following graphs

illustrating the spectral signature averaged in the whole camp area and the whole non camp area, suggests that higher separability occurs in these bands.
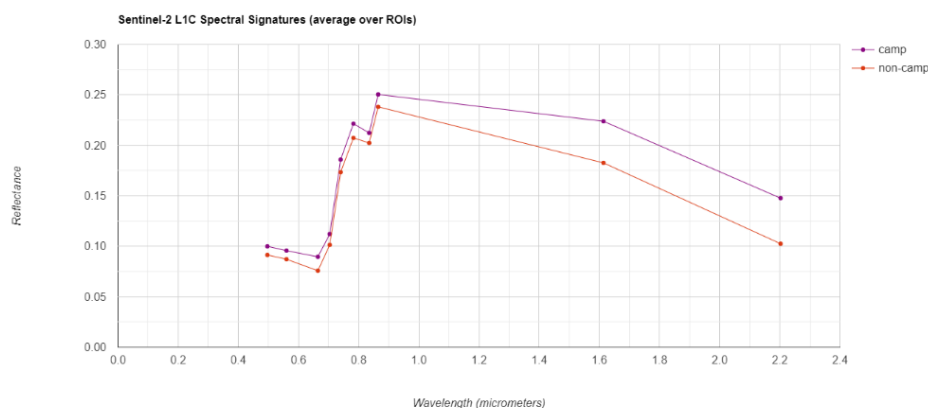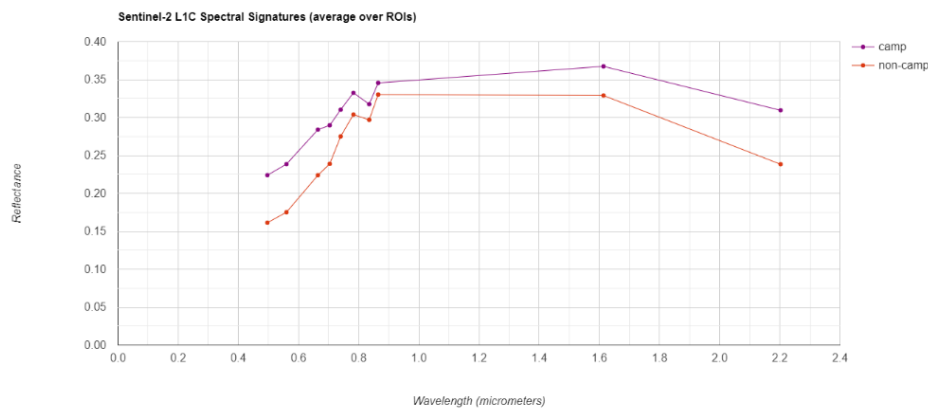

Figure 3.42 Spectral distribution of IDP camp and non-camp areas.

This selected combination of input features provided the following accuracy metrics expressed as error matrices, while the accuracy metrics for the remaining combinations of input features are illustrated in the Annexes.

VALIDATION ACCURACY

		Predicted			
	Iraq	**2019**			
	Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS				
		Non-camp	Camp		
	Non-camp	1160605	6125	1166730	
Actual	Camp	18100	30398	48498	
	Overall	1178705	36523	1215228	
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.005249715	0.015355835	0.994750285	0.984644165
	Camp	0.373211266	0.167702544	0.626788734	0.832297456
	Overall	0.980065469			

Figure 3.43 Combination of bands with the highest overall validation accuracy metrics for 2019 classification of the Iraq case study.

In the following figure, the result of the 2019 classification achieving the best overall validation accuracy compared to the manually digitized data is illustrated, highlighting where commission or omission errors occur and where agreement between the two data occur.

In this specific case, IDP camps have very clear outlines being formal structured camps.
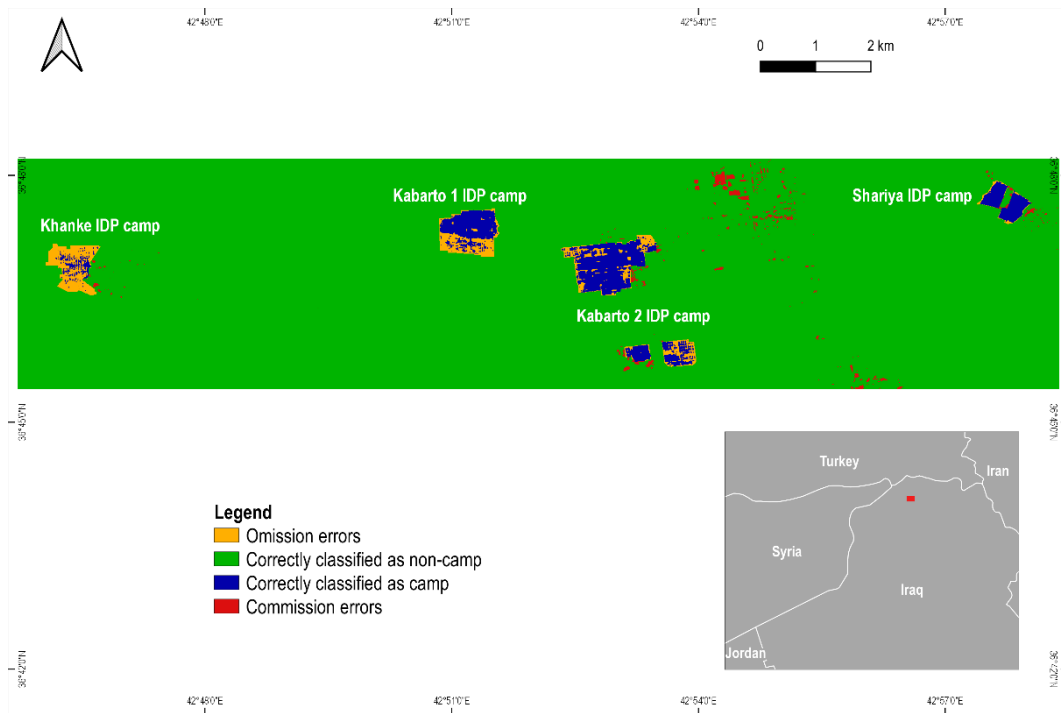
Figure 3.44 2019 IDP camps classification accuracy metrics.

In the next pages, instead:

- RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2019;
- RGB composite of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data for reference year 2019;
- Yearly median of VIIRS night-time data for reference year 2019;
- Outcome of the clustering classification for reference year 2019;
- Binary representation of the clusters, meaning that all clusters not representing the IDP camp feature have been dissolved into a single cluster and the remaining one is the cluster better representing the IDP camp feature, for reference year 2019.

Again, all classification images refer to the classification achieving the best overall validation accuracy.
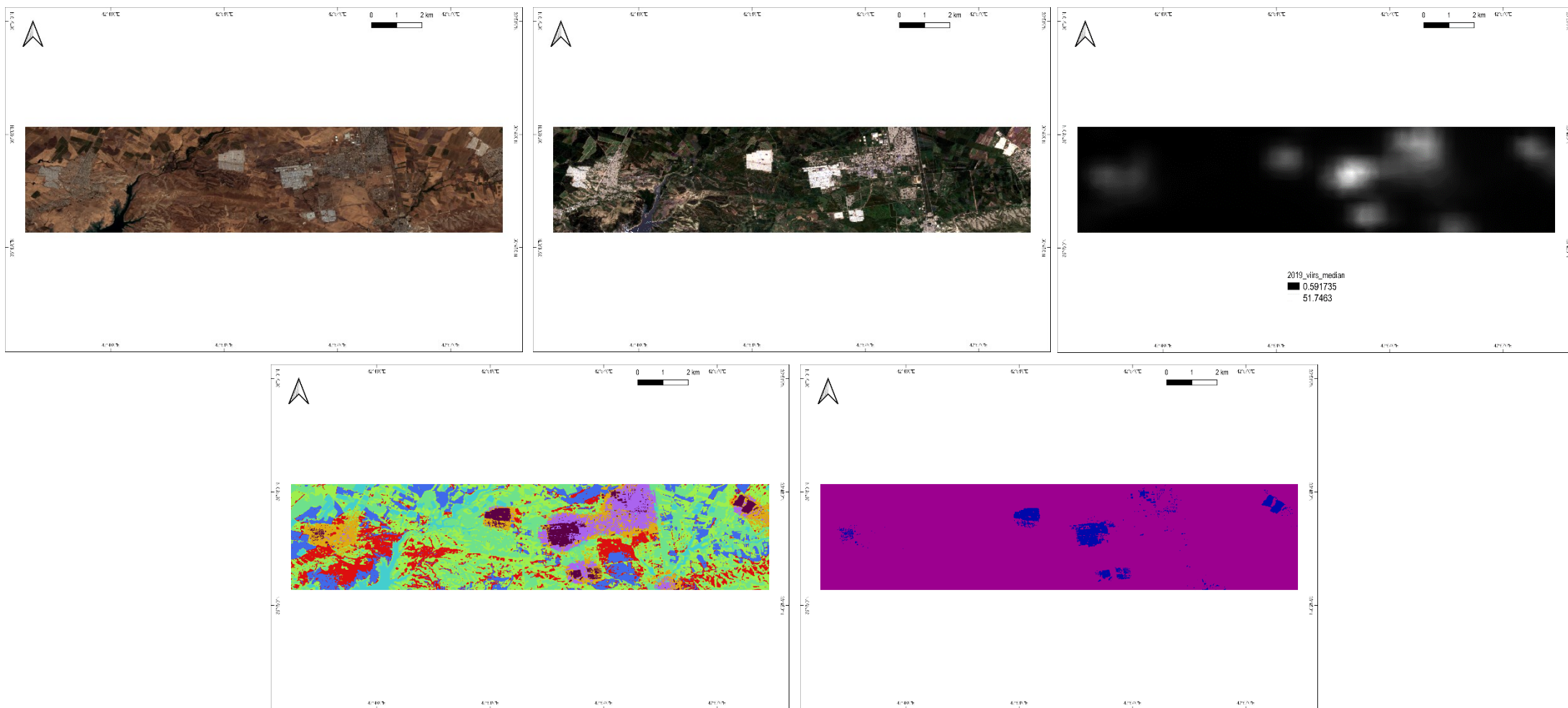
Figure 3.45 Top from left to right: RGB composite of yearly medians composites of Red, Green and Blue bands of Sentinel-2 L1C data, RGB composite of greenest composites of Red, Green and Blue bands of Sentinel-2 L1C data, yearly median of VIIRS night-time data. Bottom from left to right: clusters resulting from k-means classification, the same clusters aggregated in order to highlight IDP camp and non-camp features.

3.5.9. Discussion and conclusions

The presented series of classification exercises leads to a series of considerations:

- Sentinel-2 data offers the possibility to identify visually and to semi-automatically classify displacement settings with a measurable validation accuracy, as resumed in the following table illustrating the accuracy metrics of all four case studies for the highest overall validation accuracy achieving combination of input classification features/properties.

VALIDATION ACCURACY

Bangladesh	2019			
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS				
	Omission	*Comission*	*PA*	*UA*
Non-camp	0.025592687	0.100917221	0.974407313	0.899082779
Camp	0.281131945	0.083838377	0.718868055	0.916161623
Overall	0.902836105			
Uganda	**2017**			
Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12				
	Omission	*Comission*	*PA*	*UA*
Non-camp	0.020113421	0.071769646	0.979886579	0.928230354
Camp	0.706446997	0.389826569	0.293553003	0.610173431
Overall	0.913409432			
Ethiopia	**2019**			
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B8A, B11, B12, VIIRS				
	Omission	*Comission*	*PA*	*UA*
Non-camp	0.033928557	0.069395821	0.966071443	0.930604179
Camp	0.44901589	0.277354754	0.55098411	0.722645246
Overall	0.908681975			
Iraq	**2019**			
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS				
	Omission	*Comission*	*PA*	*UA*
Non-camp	0.005249715	0.015355835	0.994750285	0.984644165
Camp	0.373211266	0.167702544	0.626788734	0.832297456
Overall	0.980065469			

Table 3.2 Validation accuracy metrics for all case studies.

- The achieved validation accuracy varies according to the type of displacement setting and surrounding environment. In addition, every context requires a different set of input classification features/properties. In the case of Uganda, with the lowest achieved overall validation accuracy among all case studies, the refugee settlements are separable from the remaining classes in that they are highly characterized with bare soil. Nevertheless, this leads to a high degree of commission error, identifying as refugee settlements all the surrounding bare soil areas. On the other hand, in the Iraq case study, IDP camps are constituted mostly by white plastic sheets covered tents and tinned structures. Consequently, the displacement setting is very

well separable from the surrounding context and even from the surrounding urban areas.

The season of selection of images also influences the resulting classification. In the case of Uganda, where the settlement is mostly characterized by bare soil, the highest separability occurs recurring to the greenest pixel compositing techniques, in that every pixel of the classified image will be taken at the yearly peak of NDVI and surrounding vegetation will be clearly distinguishable from the soil cleared in order to host the refugee settlement. The same occurs in a very arid region such as the one of the Ethiopia case study.

- VIIRS night-time data provided an increase in the accuracy for three out of four case studies. In the case of Uganda, the data were not significant.

- Sentinel-2 data can be used to monitor periodical size evolution of displacement settings, such as in the case of the Kutupalong refugee camp in Bangladesh.

- Finally, the measures of accuracy metrics against a validation dataset in the case of displacement settings is highly challenging. Indeed, these types of features often do not have clear and easily distinguishable outlines, unless in the case of very structured formal camps, such as in the Iraq case study. To decide where to trace a line to separate the displacement camp from everything else is often a subjective choice of the image analyst.

Chapter 4

4. Cropland abandonment as an indicator of migration and displacement

When exploring applications of available data and methods in the context of displacement, indirect approaches can also be considered. Displacement is often induced by a series of factors or it is an inductor itself of a series of potential consequences (conflict, natural disasters, food insecurity, etc.). Consequently, by monitoring driving or impacted factors as indicators for displacement, it is possible to:

- Obtain indirect information on displacement;
- Monitor displacement;
- Predict displacement.

One of the driving causes of displacement is the impact of conflict on food security, leading to consequences such as cropland abandonment. This land-use change can potentially be monitored from remotely sensed data, including high-resolution data, which can give a finer level information about where this change occurs. Limitations and challenges exist.

4.1. Literature review

The topic of cropland abandonment mapping has been covered by the literature. Different approaches have been explored, ranging from visual interpretation of imagery to semi-automatic information extraction. A commonly agreed method for monitoring cropland abandonment is still lacking, especially for large scale applications.

Difficulties in monitoring cropland abandonment from remotely sensed data include the spatial dispersion of this land cover change and the challenging spatial similarity of cropland and cropland change with other features such as grasslands, fallow fields, etc. Furthermore, cropland mapping requires a sufficient amount of imagery in order to capture the dynamic temporal spectral behaviour of crops and to enable separation with other classes (Yin et al., 2020).

Operational services, such as the World Food Programme of the United Nations and the Copernicus Risk and Recovery Mapping (RRM) Service, have been already called to try answering this task.

In 2020, WFP worked on preparing cropland loss maps between 2016 and 2019 for the Mopti region, in Mali. The context of Mali facing an increasing presence of armed groups and violence outbreaks, led to loss of livelihoods and difficulties in accessing and cultivating fields for displaced populations. The methodology followed by WFP consisted in visually inspecting, comparing, and classifying multi-temporal composites of high-resolution satellite imagery, specifically Sentinel-2 data. The data was processed in order to obtain for each year interested by the monitoring a 3 bands stack where each band is the per-pixel composite of the maximum NDVI over a specific agricultural period. Specifically, the agricultural periods are the plowing, the growing and the harvesting. These multi-temporal stacks have then been used to visually compare different years and assign to each populated place in the study area a slight increase, no change, slight decrease, medium decrease and severe decrease in the cropland change. The final output is the map represented in the following figure (WFP, 2020).
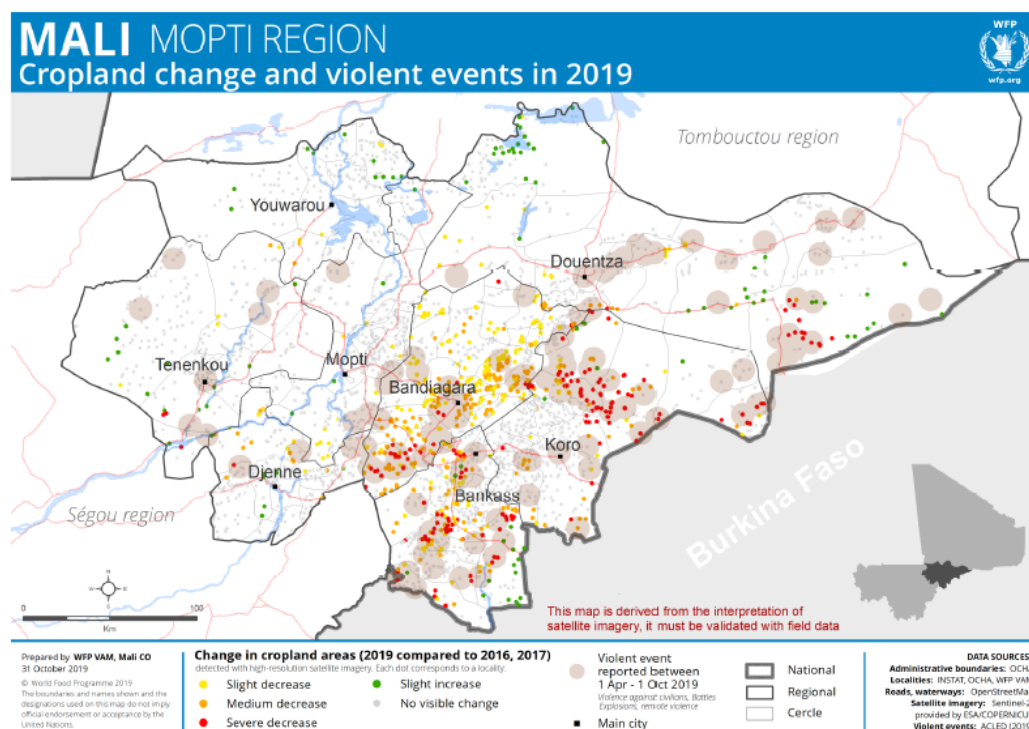
Figure 4.1 Output of WFP work in mapping cropland change after conflict events in Mopti region (WFP, 2020).

On the same year, Copernicus Risk and Recovery Mapping (RRM) Service was called by WFP itself to perform the same kind of analysis in the Borno state in North-Eastern Nigeria between 2010 and 2019. The objective was to estimate the impact of conflict and to estimate the population affected by cropland loss. In this case, availability of imagery was more challenging and required to recur to a wider range of satellite sensors to respond to the requested monitoring time range. The methodology applied was aligned with the one applied by WFP. For each settlement, cropland change classification was performed visually. The identified classes are significant decrease, medium decrease, slight decrease, no change, slight increase, medium increase, and significant increase. The different sensors resolutions were one of the stated reasons that prevented the application of a semi-automated change detection. The final product is a series of maps per Area of Interest (AOI) and a general one represented in the following figure.
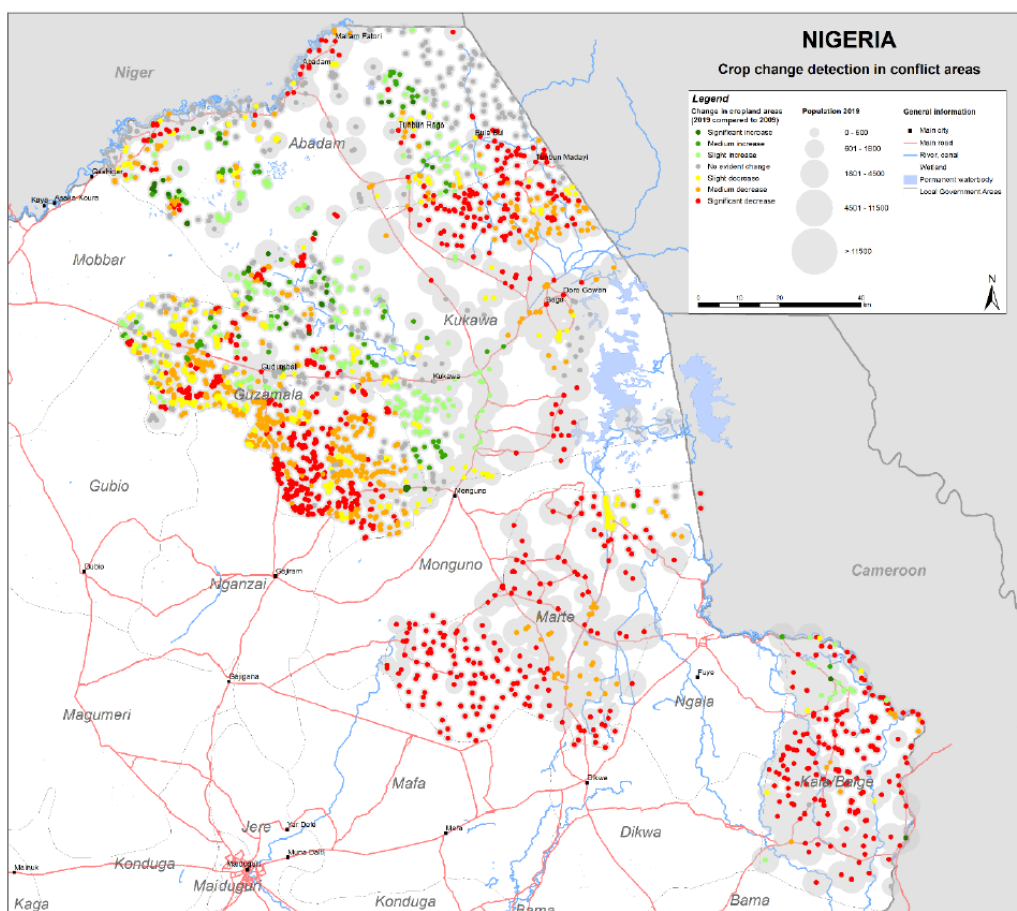
Figure 4.2 Output of Copernicus work in mapping cropland change after conflict events in North-Eastern Borno state (Copernicus, 2020).

The limitations identified in these two approaches consist in:

- The lack of a quantitative and measurable output in terms of cropland loss;
- The cropland loss output is not explicitly spatially located but generally assigned to the populated place;
- The considerable human effort and time consumed in visually interpreting a large number of villages points (more than 2000 in both cases).

Yin et al. (2020) illustrated a semi-automated method for mapping the extent and the time of cropland abandonment using the entire availability of Landsat time series, tested on 14 heterogeneous study sites. In their approach, the starting point was a yearly per-pixel classification of active cropland and non-cropland using annual summary statistics using multi-years training data identified on stable areas across years, minimizing the input and labelling of training data. The classification was performed for each study site and each year separately. This allowed not to assume that reflectance behaviour and phenology of cropland remain similar among years. The land use change was then classified according to the multi annual per-pixel evolution to distinguish between what is an actual abandoned cropland, stable cropland, non-cropland, and fallow fields (Yin et al., 2020).

4.2. Multi-temporal supervised classification in Google Earth Engine

In the context of a research collaboration and a research stay with the Z_GIS department of geoinformatics at the University of Salzburg, the possibility and extent to which an automation of the approaches followed by the WFP and Copernicus is possible is explored. The main objective is to obtain a more quantitative result in the identification of abandoned cropland and to use the output as an indicator of potential displacement.

4.2.1. Study area

It was decided to implement the analysis on the same area of interest covered by the Copernicus RRM activation in 2020, in order to have access to the already existing data about the area of study and the population sites and to have a comparative product, even though mostly qualitative. The Copernicus activation focused on five areas of interest (AOI) in Borno state in north-eastern Nigeria, west of Lake Chad, represented in the figure and listed in the table.
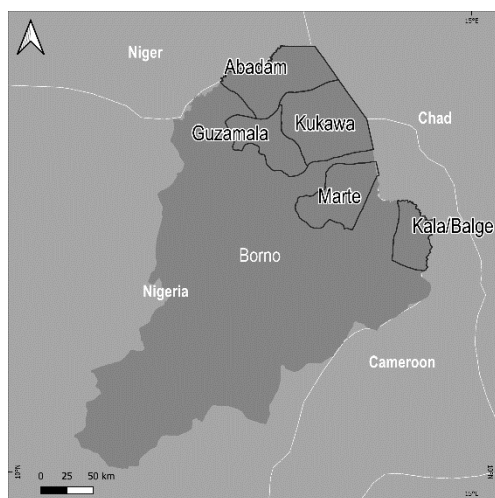
Figure 4.3 Areas of interest (AOI).

AOI No.	Name	Area [km^2]
1	Abadam	2502
2	Guzamala	3153
3	Kukawa	3989
4	Marte	4867
5	Kala Balge	1883

Table 4.1 Areas of interest.

4.2.2. Context

This area of Nigeria witnessed a dramatic conflict which originated in 2009 with the insurgence of Boko Haram and that led to an outbreak of the conflict in 2014, especially in the north-eastern states of Nigeria, including the Borno state (Copernicus, 2020). The increased security threats limited the access to the land surrounding the villages in the affected areas. This induced a significant loss in cropland which, consequently, impacted population itself, leading to a highly insecure food situation. An increased number of villages and towns underwent attacks and led the population to flee in other nearby towns. Out of them, the well-known refugee town of Rann, in the Kala Balge region. The security threats strongly limited the access to the area and consequently limited the availability of field collected data and reliable information, leaving remotely assessment as one of the few options to gain situational information on the area.

4.2.3. Tools and methods

In the context of this research, the identification and mapping of abandoned cropland in a semi-automated approach was achieved through the classification of active cropland in two reference years (pre- and post-conflict) and the delineation of changed cropland status by means of the difference between the obtained yearly cropland classification masks. To this end, multi-temporal pixel-based supervised classification is the main tool explored.

The pixel-based classification was mostly performed in the Google Earth Engine (GEE) platform, and the developed code is available in the Annexes. Preliminary analysis steps were performed recurring to other software.

The pixel-based classification was performed recurring exclusively to high-resolution satellite imagery, specifically Landsat-8 (30 meters resolution) for years before 2015 (included) and Sentinel-2 (10 meters resolution) for years from 2016 to 2020. Landsat-8 imagery was resampled to 10 meters resolution using a bilinear interpolation method in order to integrate it with Sentinel-2.

First tests of classification were performed recurring to the same multi-temporal composites stacks used during the Copernicus activation. Since the three vegetation periods composites were not successful in providing an acceptable cropland classification, it was decided to investigate different multi-temporal stacks of bands as input properties/features of a classification, as literature itself suggests for this type of classifications (Yin et al., 2020). Monthly composites of maximum NDVI were generally used when performing the reported analysis.

Clouds represented a big challenge in performing multi-temporal analysis, especially in this region where rainy months strongly impact image availability and usability even when considering month long composites. The use of maximum NDVI composites compensates the impact of clouds, but still some cloudy images ended up being integrated in the computation and hence impacted some portions of

the images. In addition, recurring to maximum NDVI composites lead to classification issues especially along water bodies, where vegetated pixels are picked-up when creating composites.

The 2010 reference year used by the Copernicus activation as pre-event year was highly challenged by the spare and diverse satellite sensors availability, as stated in the Copernicus methodology. For this reason, 2013 was used as fallback year to compensate data availability (Copernicus, 2020). Anyhow, following previous considerations on cloud coverage, in the presented analysis it was decided to perform a cropland change classification between years 2014 and 2019. The year 2014, indeed, allows to access a more stable Landsat-8 time series of imagery compared to 2013 and it is not expected to have significant difference compared to year 2013 used also in Copernicus activation. The month of August 2014 had to be compensated with imagery from August 2015 because of excessive cloud impact. Year 2019 was instead accepted as post-event reference year, given larger availability of data in the Sentinel-2 repository.
The following scheme resumes the applied methodology.
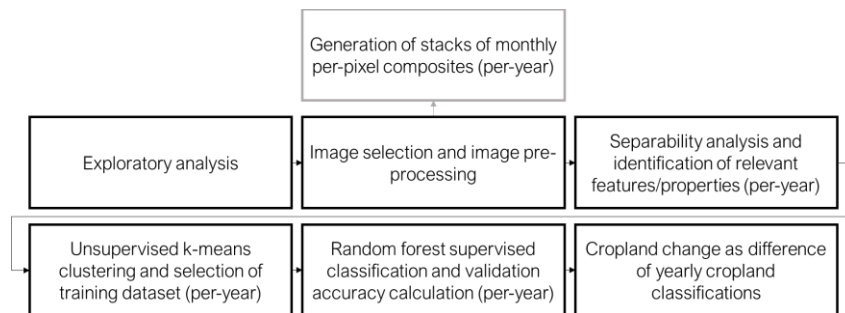
Figure 4.4 Methodology scheme.

4.2.4. Exploratory analysis

In order to identify the optimal method to approach the cropland classification, an exploratory analysis of one of the areas of interest and of the available data was initially performed. The main aim was to identify first visually and then from a spectral point of view the differences between two different years in the cropland status and between the two classes to be identified (cropland and non-cropland) in the same year in the selected AOI. The selected AOI for the exploratory analysis was Kala Balge, since it was the one with the largest degree of visually identified decrease in cropland area during the Copernicus RRM activation. Major staples in the area include cereals such as maize, millet and rice, whose seasonal calendar in Northern Nigeria spans from planting in May and harvesting in October (FEWS NET, 2018). Specifically sweet corn is planted in July and harvested in October (FAO, 2021). The crop calendar was identified by Copernicus as follows:

- Vegetation period 1: Plowing from 15th June to 1st August;
- Vegetation period 2: Growing from 2nd August to 1st September;
- Vegetation period 3: Early stage of harvesting from 2nd of September to 15th October (Copernicus, 2020).

Difference between years

Copernicus and WFP worked on visual interpretation comparing RGB synthesis of three band stacks per each year in order to identify change in cropland. Each yearly stack is composed of three bands, where each band represents the per-pixel composite of the maximum NDVI over one of the vegetation periods listed above. For Kala Balge, these stacks were recreated in Google Earth Engine and the results are showed in the following images.
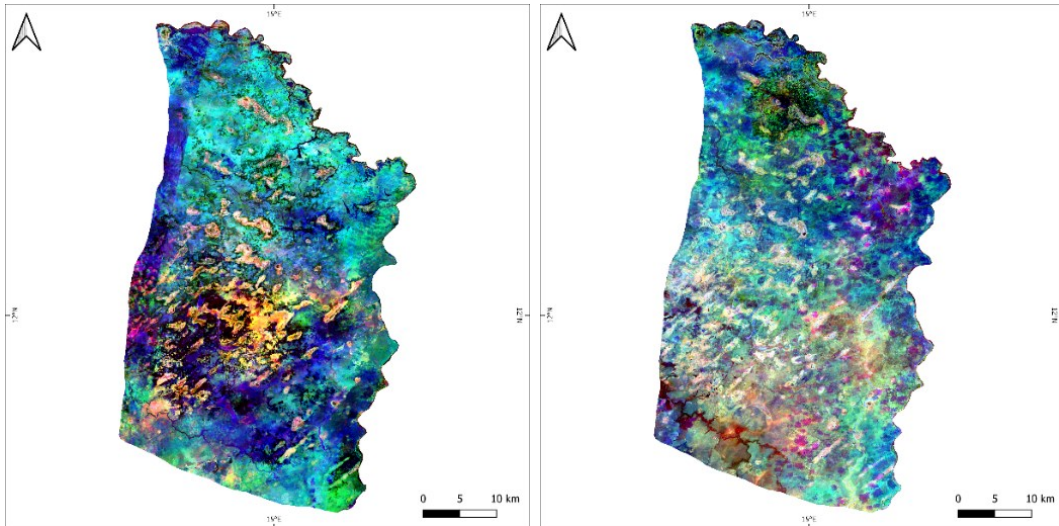
Figure 4.5 RGB synthesis of per-pixel maximum NDVI composites of three vegetation periods over Kala Balge AOI for reference year 2013/2014 using Landsat8 imagery (left) and 2019 using Sentinel-2 imagery (right). Original satellite imagery in Top of Atmosphere (TOA) reflectance values.

The following images show instead two per-pixel composites of maximum NDVI exclusively over the month of October for the two reference years 2014 and 2019. In these composites the difference between the two images in the cropland distribution is more striking, being October the harvesting month. The consequence is that harvested fields exhibit lower NDVI (darker in the images) compared to the surrounding vegetation.
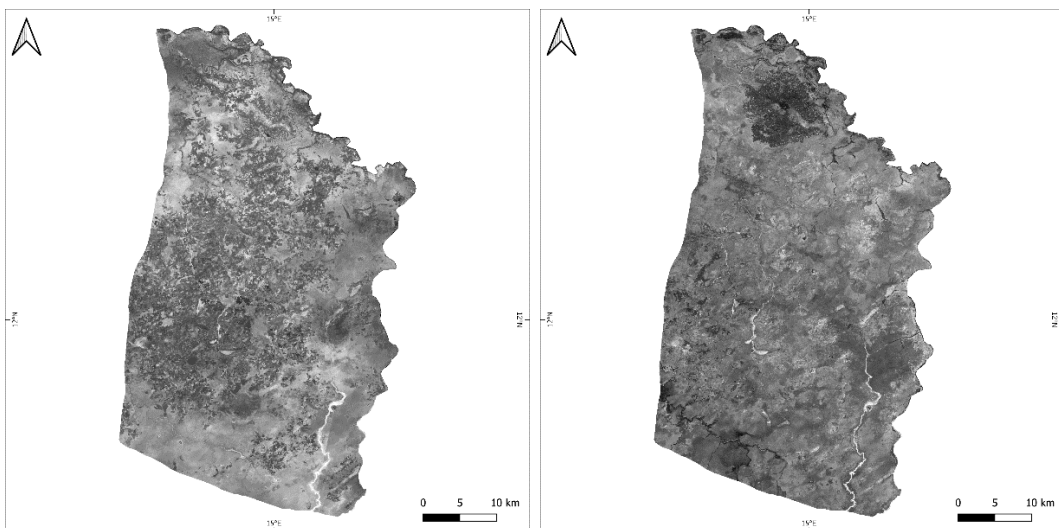
Figure 4.6 Per-pixel maximum NDVI composites of October over Kala Balge AOI for reference year 2014 using Landsat8 imagery (left) and 2019 using Sentinel-2 imagery (right). Original satellite imagery in Top of Atmosphere (TOA) reflectance values.

93

A close-up over the town of Rann shows the striking differences in the cropland distribution between the two years. Rann, also called a refugee town, hosted an increasingly larger amount of Internally Displaced People (IDP) from the neighbouring villages over the years of conflicts, and this is also captured by the expanding cropland area around the town itself. From these images, the difference between a cropland and the surrounding environment is highlighted by its darker colour in the image, as already stated reflecting the lower NDVI seemingly linked to the harvest of the crop itself.
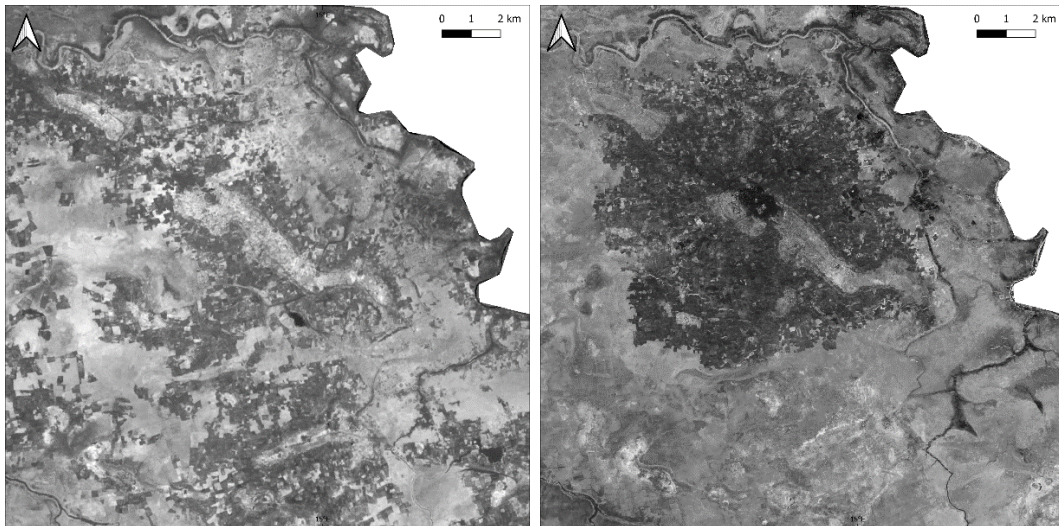
Figure 4.7 Per-pixel maximum NDVI composites of October over the town of Rann for reference year 2014 using Landsat8 imagery (left) and 2019 using Sentinel-2 imagery (right). Original satellite imagery in Top of Atmosphere (TOA) reflectance values.

A close-up over towns that were instead identified as destroyed by Copernicus activation in other areas of Kala Balge shows the difference in the cropland distribution and likely abandonment between the two years.
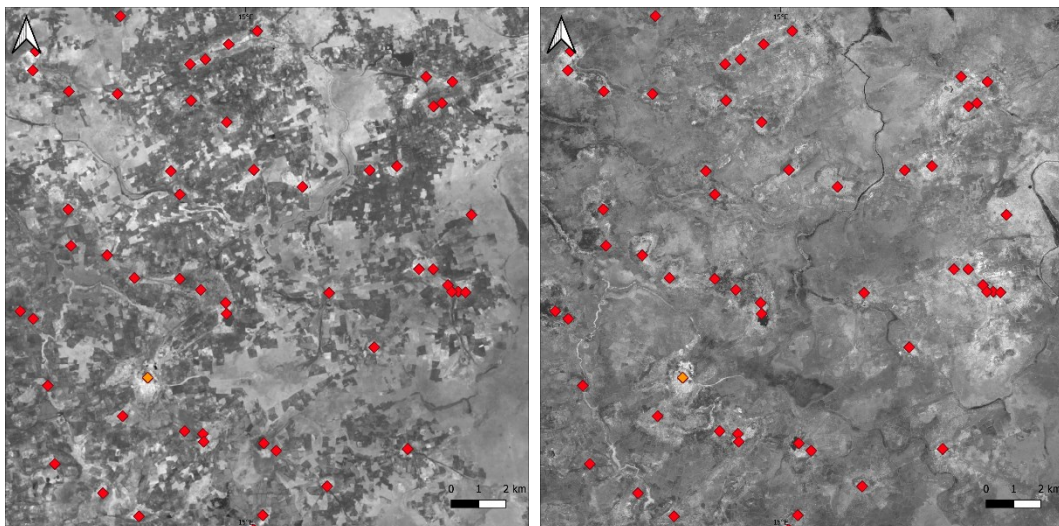
Figure 4.8 Per-pixel maximum NDVI composites of October over destroyed and partially functional towns (red and orange dots) for reference year 2014 using Landsat8 imagery (left) and 2019 using Sentinel-2 imagery (right). Original satellite imagery in Top of Atmosphere (TOA) reflectance values.

In the following images, the year-by-year evolution over the town of Rann is showed. Images are this time three bands synthesis where each band is a per-pixel

composites of maximum NDVI over the entire months of August, September and October. For 2014 Landsat-8 series were used, for remaining years Sentinel-2 series were used, in both cases original satellite imagery was in Top of the Atmosphere (TOA) reflectance values. Crop field stand out in dark colours. The progressive increase in cropland area around the city of Rann is again evident.
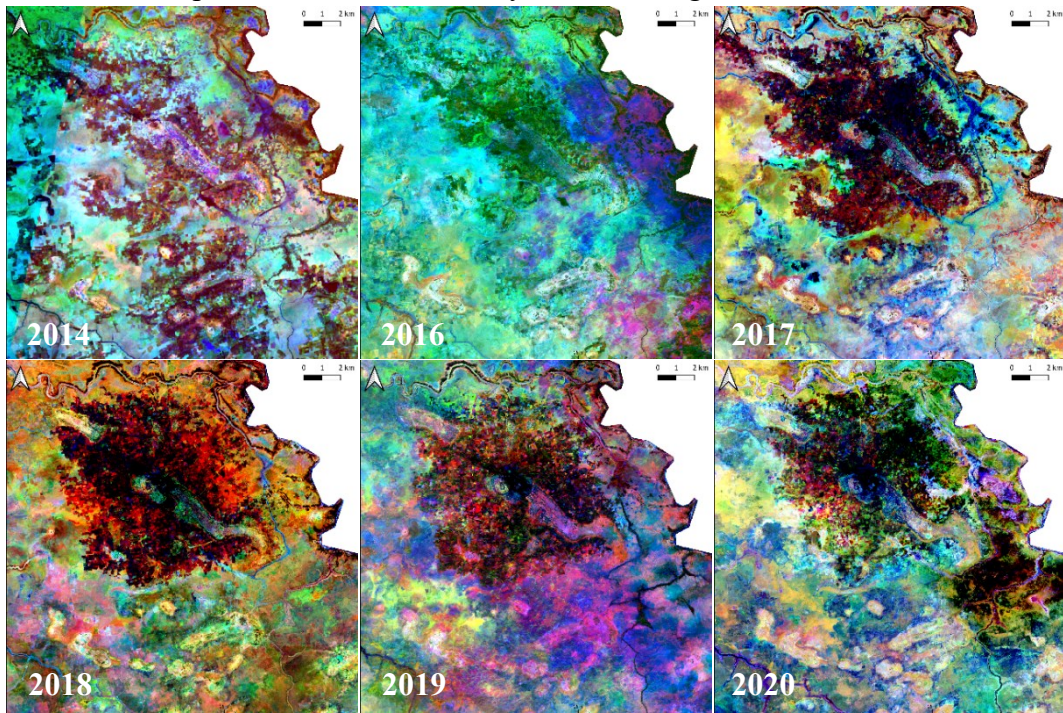
Figure 4.9 Yearly RGB synthesis of per-pixel maximum NDVI composites of August, September and October over the town of Rann for 2014 using Landsat8 imagery Sentinel-2 imagery for remaining years. Original satellite imagery in Top of Atmosphere (TOA) reflectance values.
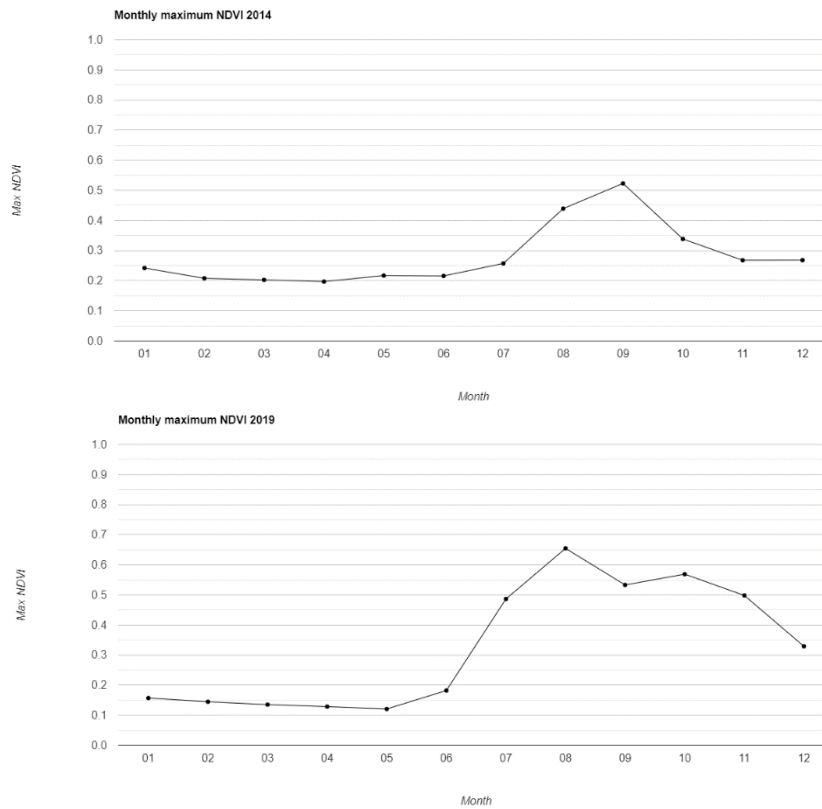
Figure 4.10 Charts showing average maximum NDVI per month in 2014 and 2019 over the same sample cropland patches.

95

The behaviour of crop fields that visually seemed to be active in 2014 and not active anymore in 2019 was further analysed by investigating the multi-temporal spectral evolution of the NDVI of some test cropland patches. To this end, stacks of per-pixel maximum NDVI composites per each month of the two reference years 2014 and 2019 were generated recurring to Landsat-8 Bottom of the Atmosphere (BOA) reflectance imagery for 2014 and Sentinel-2 Bottom of the Atmosphere (BOA) reflectance imagery for 2019. A sample of cropland patches that "visually" look active in 2014 while inactive in 2019 was selected in areas cleared by clouds disturbance. The multi-temporal maximum NDVI signature of these sample fields was plotted for both 2014 and 2019. The resulting charts are showed in the previous figure, where the average over all sample crop patches of maximum NDVI values per month is presented for both years. In 2014, a drop in the NDVI signal is visible in the month of October, seemingly linked to the harvesting, while in 2019 the more regular curve resembles the ones of herbaceous vegetation.

Difference between classes

To understand whether the two classes of interest (cropland and non-cropland) are spectrally separable in the selected images, a preliminary unsupervised isodata clustering and separability analysis of classes were performed. The input data are:

- 2014 (August 2014 compensated with August 2015 images) stack of per-pixel maximum NDVI composites per month, resulting in a 12 bands stack where each band is composed by the max NDVI per month per-pixel. Landsat-8 BOA reflectance values are used as input for the composites calculation in GEE.
- 2019 stack of per-pixel maximum NDVI composites per month, resulting in a 12 bands stack where each band is composed by the max NDVI per month per-pixel. Sentinel-2 BOA reflectance values are used as input for the composites calculation in GEE.

An isodata clustering algorithm with minimum number of clusters of 2 and maximum number of clusters 11 was run on both stacks of images, taking into account all twelve bands. For year 2014, the isodata classification resulted in 6 classes, of which three visually aligned with the active cropland with some relevant commission error. For year 2019, the isodata classification resulted in 8 classes, of which two visually aligned with the active cropland with some relevant commission error. The resulting clusters for both years identifying the active cropland are illustrated in the following figure.
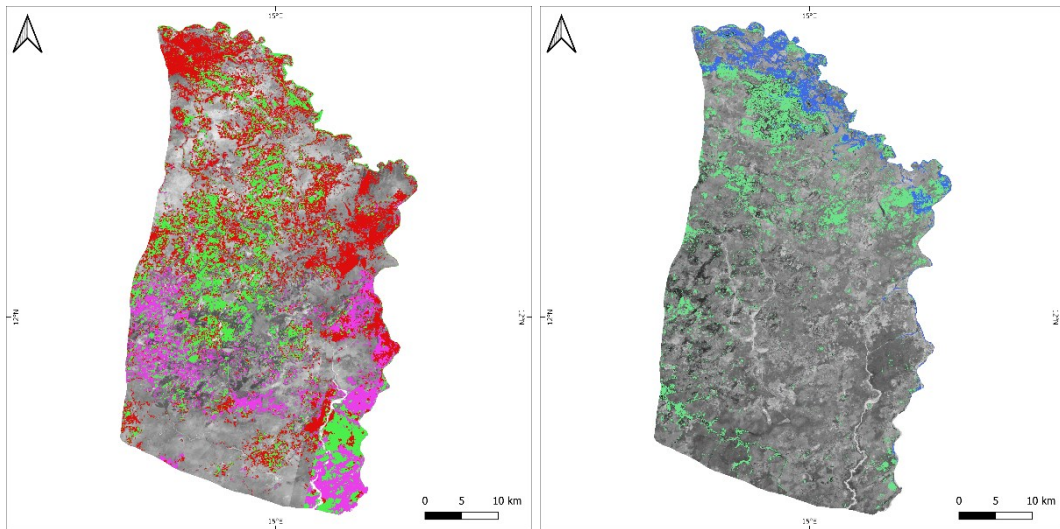
Figure 4.11 Results of isodata unsupervised clustering on 2014 (left) and 2019 (right) multi-temporal stacks of maximum NDVI composites. Original satellite imagery in Bottom of Atmosphere (BOA) reflectance values.

A number of Region of Interests (ROIs) were identified within the clusters aligning with cropland and non-cropland and labelled according to their class (cropland and non-cropland) in order to assess the spectral separability of the clusters in every band constituting the stack, consequently for each month of the year. The main objective is to understand whether all input features/properties, hence all months of a year, are relevant in separating between the two cropland and non-cropland classes. The charts in the next page show the results of the separability analysis for 2014 and for 2019 for the two classes for the selected ROIs. Both in terms of multi-temporal maximum NDVI signature averaged across the ROIs and in terms of scatter plots. Only most relevant scatterplots are illustrated together with one examples of scatter plots of two months in which separability is more challenging.

From the analysis and from the figures, it is evident how not all months are relevant when trying to separate the cropland from the non-cropland class. In 2014, the months of August, September and October are the ones with the higher spectral separability in terms of maximum NDVI. In 2019, the window of higher separability is larger, going from July to November, but higher in August, September, October and November. This analysis provides an insight on which properties/features can be more relevant when performing further classification exercises.

Statistical separability tests were also performed on the ROIs for 2014 and 2019 on the identified most separable months, hence August, September and October. Spectral separability between class ROIs were measured with the Jeffries - Matusita and Transformed Divergence separability measures, achieving respectively values of 1.99 and 1.99 for 2014 and 1.92 and 1.99 for 2019, indicating that the ROIs class pairs have a good statistical separability in the identified months.

97
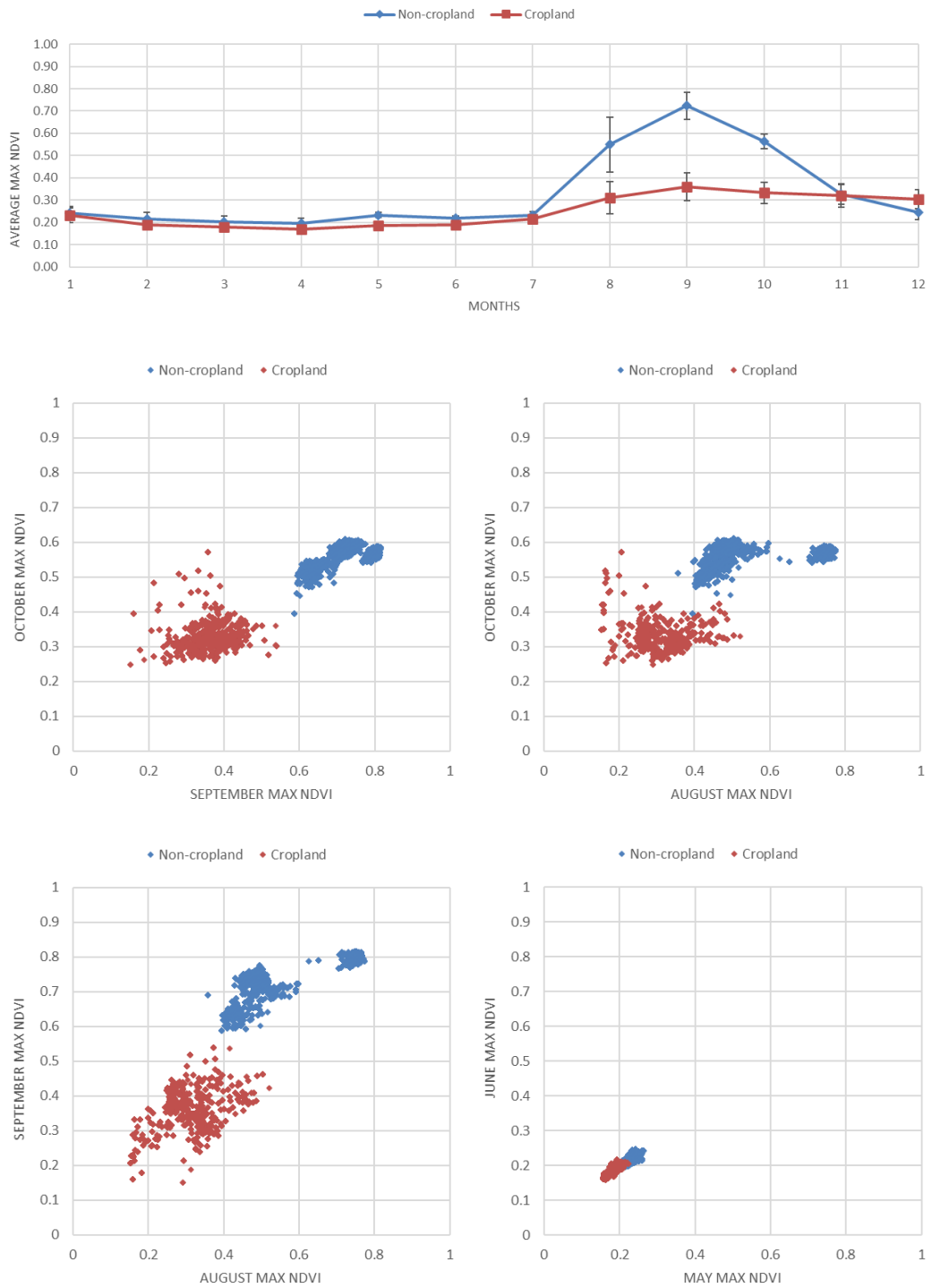
Figure 4.12 Spectral separability analysis on 2014 imagery of cropland and non-cropland ROIs.
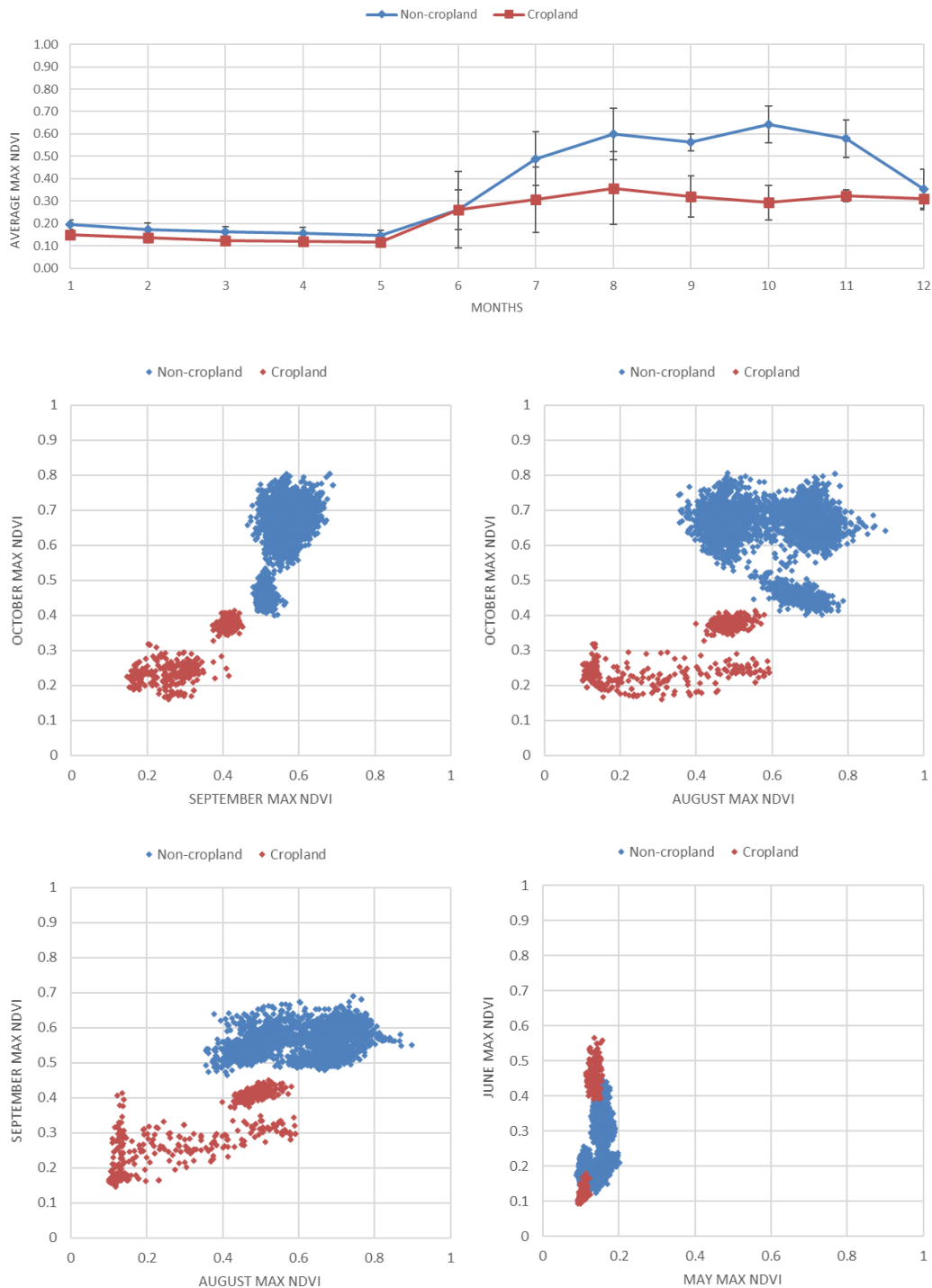
Figure 4.13 Spectral separability analysis on 2019 imagery of cropland and non-cropland ROIs.

4.2.5. Pixel-based supervised classification

The previous exploratory analysis served as a basis to identify those properties/features that are expected to maximize the separability between the cropland and non-cropland classes for a test AOI whose size allowed a simultaneous visual inspection of the results. This served as a starting point to extent the analysis on the entire study site, all areas of interest, recurring to both an

99

unsupervised clustering to first identify the training dataset and then a supervised classification, both exclusively based on the most relevant multi-temporal features.

Creation of training and validation dataset

A supervised classification, as mentioned in the previous chapter, requires a training dataset and a validation dataset at least.

In order to generate a training dataset made of spectrally separable elements and that include a certain degree of automation, the unsupervised clustering approach tested on the test AOI is applied over all the areas of interest. This allows to identify clusters of spectrally separable classes, identify the clusters that physically represents the classes of interest (cropland and non-cropland) and sample within these clusters the desired amount of training data. The potential noise in the outcome training dataset will be dampened by the following supervised random forest classifier, as it will be explained in the following section.

In order to generate a training dataset of separable samples, new stacks containing only relevant properties/features derived from the exploratory analysis were generated for the totality of the AOI. This resulted in a three bands stack for 2014 constituted by per-pixel maximum NDVI composites of August, September and October and in a three bands stack for 2019 constituted by per-pixel maximum NDVI composites of August, September and October using BOA reflectance values as input for the composites calculation in GEE both for Landsat-8 (2014) and Sentinel-2 (2019). Both stacks were again generated in Google Earth Engine environment. These stacks are illustrated as RGB synthesis in the figures in the next page, where also problematic areas in terms of cloud cover are highlighted. These stacks were then used as inputs for unsupervised clustering of the images, this time recurring to a k-means clustering algorithm since it is the one available in the GEE platform. Both years underwent unsupervised clustering with a varying number of clusters until the classes of interest were approximately represented. These clusters were then used to label a dataset of 50 random points per individual AOI as cropland and non-cropland both for 2014 and for 2019. These random points were then buffered in 10mx10m squares. This final dataset represents the training dataset used for the pixel-based supervised classification. Jeffries - Matusita and Transformed Divergence separability measures achieved respectively values of 1.22 and 1.34 for 2014 and 1.26 and 1.67 for 2019, indicating that the training dataset could potentially be improved in its statistical separability.

The validation dataset was instead created independently by visually interpreting a series of randomly generated points by two independent remote sensing experts at the University of Salzburg. Approximately 260 randomly generated points, approximately 50 per individual AOI, were visually inspected and labelled according to their land cover (cropland and non-cropland) in both 2014 and 2019 using the following class definition: active cropland are areas that shows over

the year the behaviour of typically active tilled crops, consequently manifesting a clear soil signal at some point during the year as a consequence of harvesting (Yin et al., 2020); non-cropland are areas that are not tilled, consequently not showing the behaviour of active cropland, including herbaceous vegetation, woody vegetation, non-vegetated areas, urban areas, etc.
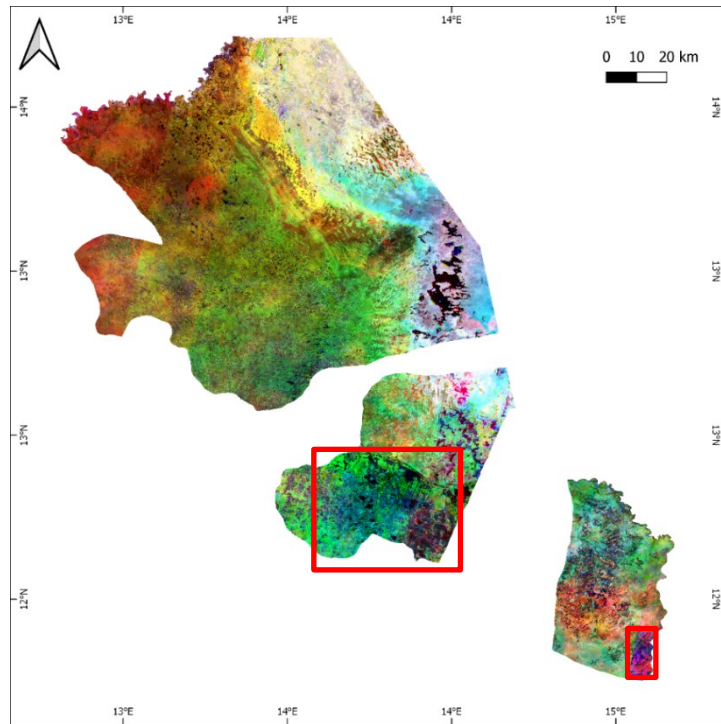
Figure 4.14 RGB synthesis of per-pixel maximum NDVI composites of August, September and October 2014. Original satellite Landsat-8 imagery in Bottom of Atmosphere (BOA) reflectance values. Cloud impacted areas highlighted by red squares (clouds impacted month is August).
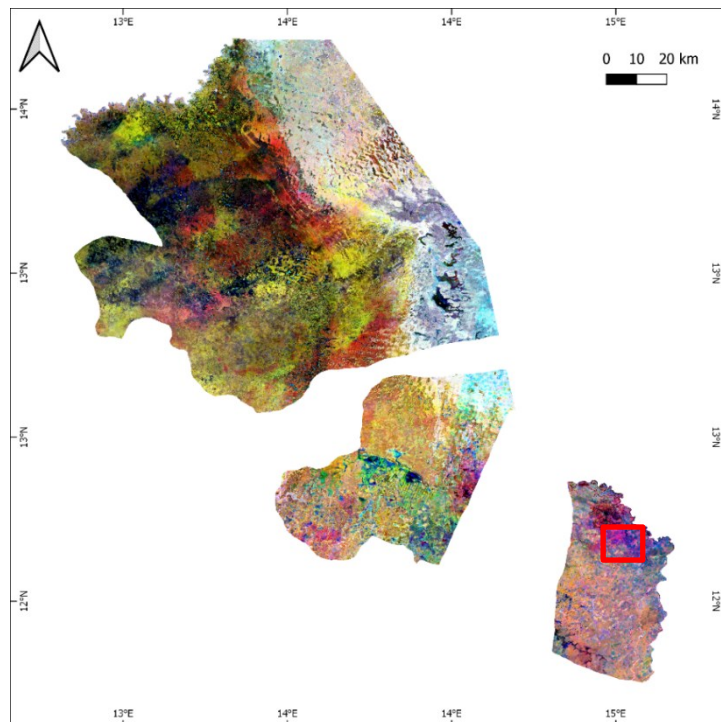
Figure 4.15 RGB synthesis of per-pixel maximum NDVI composites of August, September and October 2019. Original satellite Sentinel-2 imagery in Bottom of Atmosphere (BOA) reflectance values. Cloud impacted areas highlighted by red squares (clouds impacted month is September).

The following charts show the spectral separability of the identified training dataset in all months. The average maximum monthly NDVI of all training areas labelled with the same class is plotted. These charts confirm that the spectral separability within the training dataset is higher within the months of August, September and October for both years.
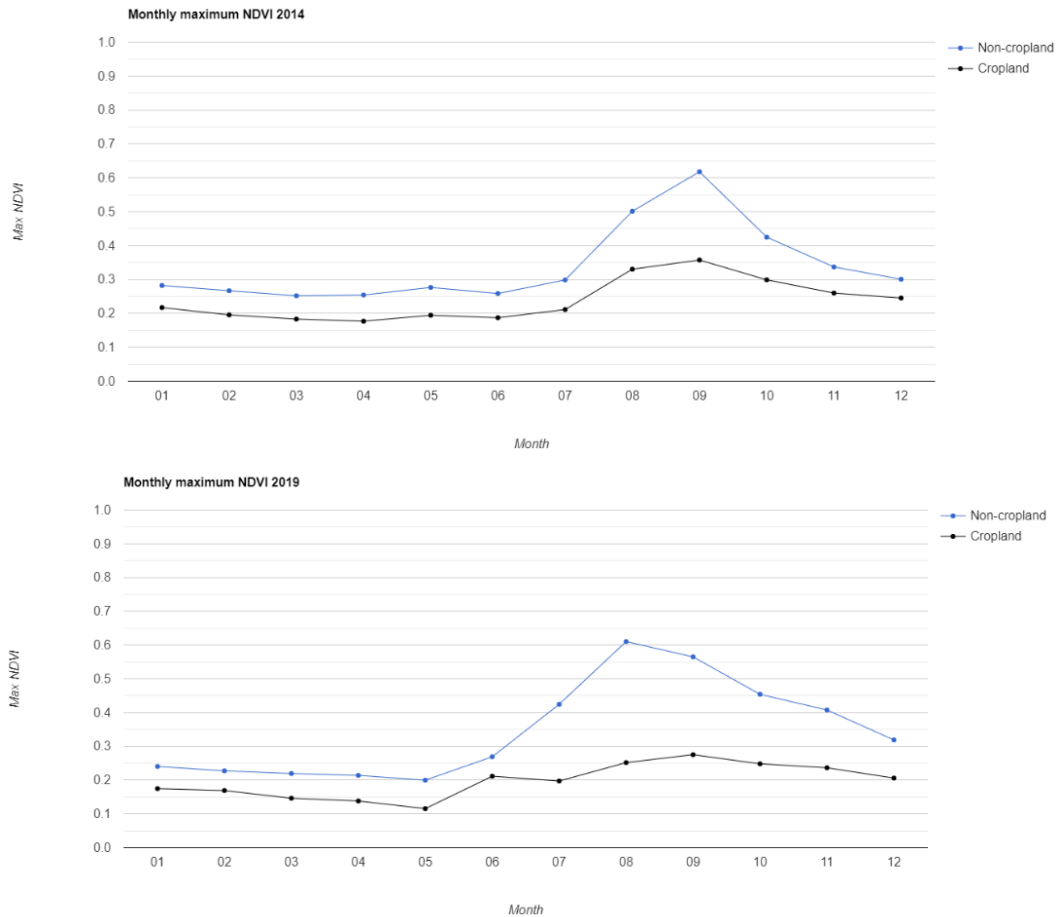
Figure 4.16 Charts showing average maximum NDVI per month in 2014 and 2019 per class of the labelled training areas constituting the training dataset.

Random forest stratified classification

The supervised classifier intended to be used was a random forest, commonly adopted in the literature for this type of classification, with 100 trees. Random forest classifier repeatedly takes shuffled subset of both the training data, by standard 50% of all pixel withing the training data, and of the input properties/features, by standards the square root of the number of features. This means that each tree of the classifier is created based on a different set of training data and features. Eventually, all the trees are summarized to a majority voting. This method makes the random forest classifier robust towards outliers in the training dataset and in redundancies in the input features. This also makes important to have enough trees. Indeed, the more input properties/features to the classifier the more important is the factor of randomization of input features in order to ensure that all features will be used to construct the internal random forest rulesets. A sufficient number of trees, meaning

that the process is repeated for as many trees, ensures that the permutation of input features and training data covers all the inputted ones.

The random forest classifier was trained and applied with a stratified approach. This means that a random forest classifier was trained and applied independently for each AOI constituting the whole study area (hence 5 classifiers where trained and run independently), selecting for each training and classification only the portion of training dataset included exclusively in the AOI of interest. The validation of the resulting classification was then again tested independently for each individual AOI using the visually labelled validation dataset. The resulting cropland classification for reference years 2014 and 2019 were then arithmetically subtracted to obtain a cropland change classification map between 2014 and 2019. All the procedure was implemented in a Google Earth Engine code showed in the Annexes.

Results

The following figures represent the resulting cropland change map as a difference of 2014 and 2019 classifications cropland masks for the whole study area. The following colour legend is followed:

- White: never classified as cropland;
- Yellow: stable cropland;
- Green: new cropland;
- Red: lost cropland.

The error matrices for individual yearly classifications for each individual AOI are reported in the Annexes. Results of the final cropland and non-cropland classifications for individual years report low and not acceptable validation accuracies metrics for almost all the AOI, especially for the cropland class. A potential reason could be the unbalanced distribution of training and validation data between the cropland and non-cropland classes, particularly with less validation samples for the cropland class itself due to minor distribution of this class across the land cover, especially in 2019.

A close-up of the results of the classification over the Kala Balge region is also reported in the following figures. In Kala Balge it can be highlighted how that the classification managed to capture the change in cropland around some areas, such as Rann, and that some commission error is included in the classification. It is evident, for instance, how often riverbeds are mistaken for agriculture, probably due to the use of maximum NDVI composites that implies the selection of vegetated pixel over a time window.

Finally, some further close-ups and comparison between classification and RGB synthesises of multi-temporal stacks of images for 2014 and 2019 are illustrated to highlight and allow further comparisons.
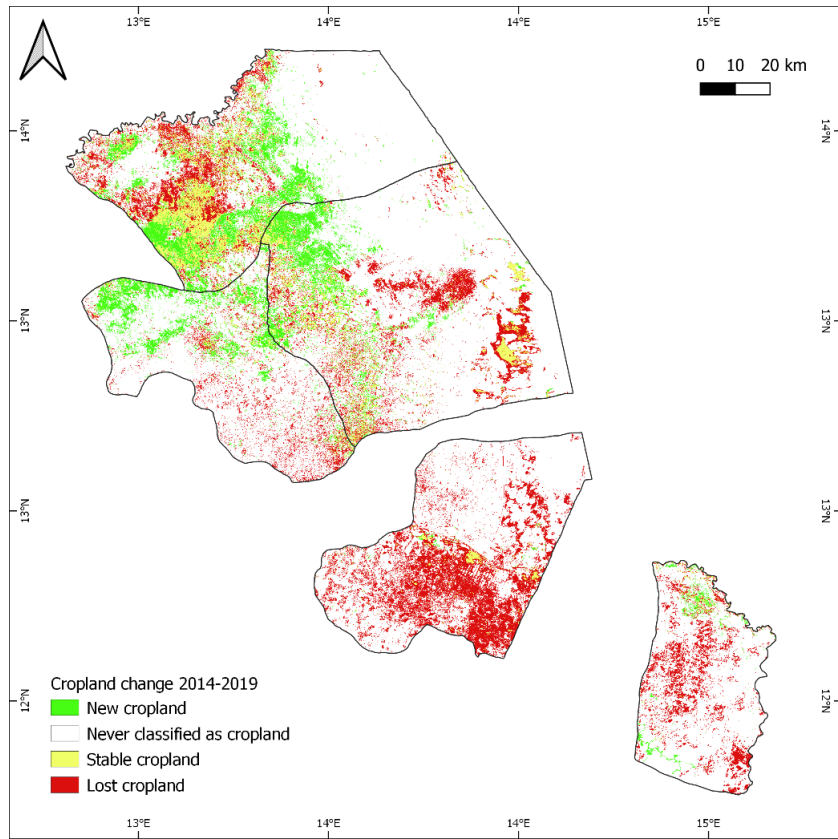
Figure 4.17 Cropland change classification between years 2014 and 2019 map over the five AOIs.
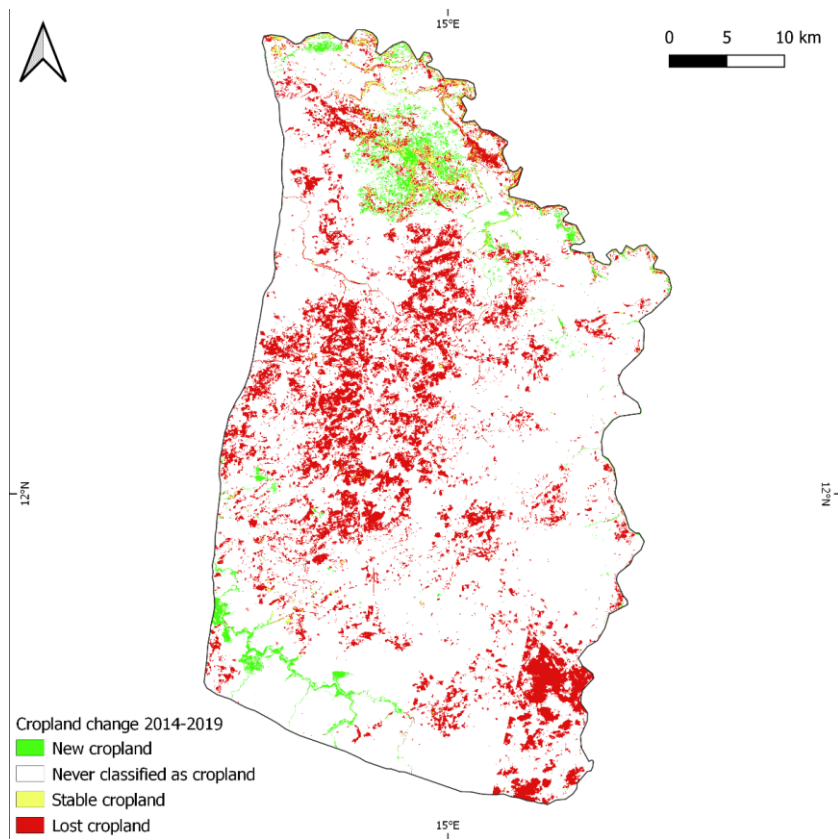
Figure 4.18 Cropland change classification between years 2014 and 2019 map over Kala Balge AOI.
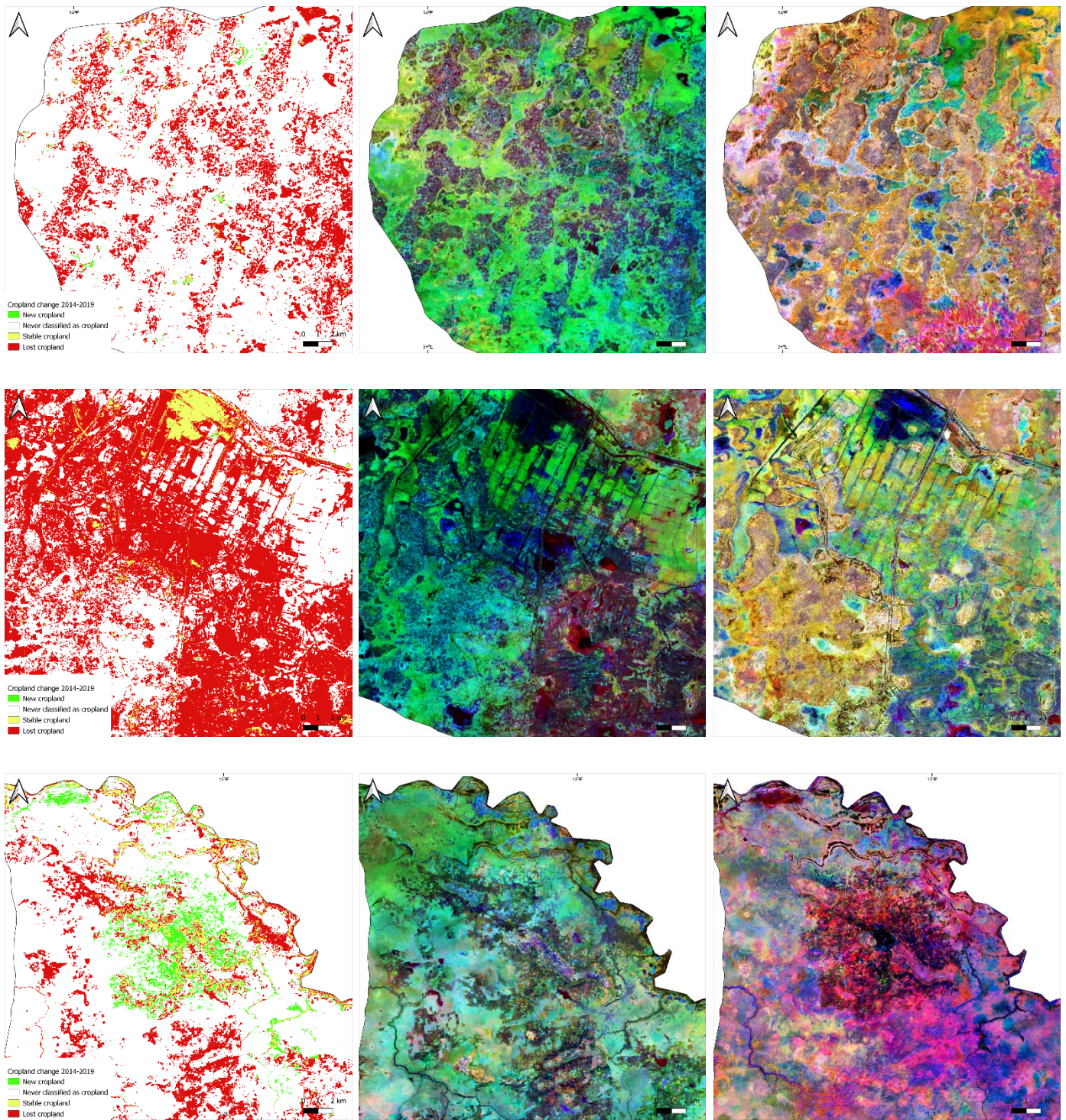
Figure 4.19 Samples close-ups. From left to right: cropland change classification between years 2014 and 2019, 2014 RGB synthesis of per-pixel maximum NDVI composites of August, September and October using Landsat8 imagery, 2019 RGB synthesis of per-pixel maximum NDVI composites of August, September and October using Sentinel-2 imagery. Original satellite imagery in Bottom of Atmosphere (BOA) reflectance values.

105
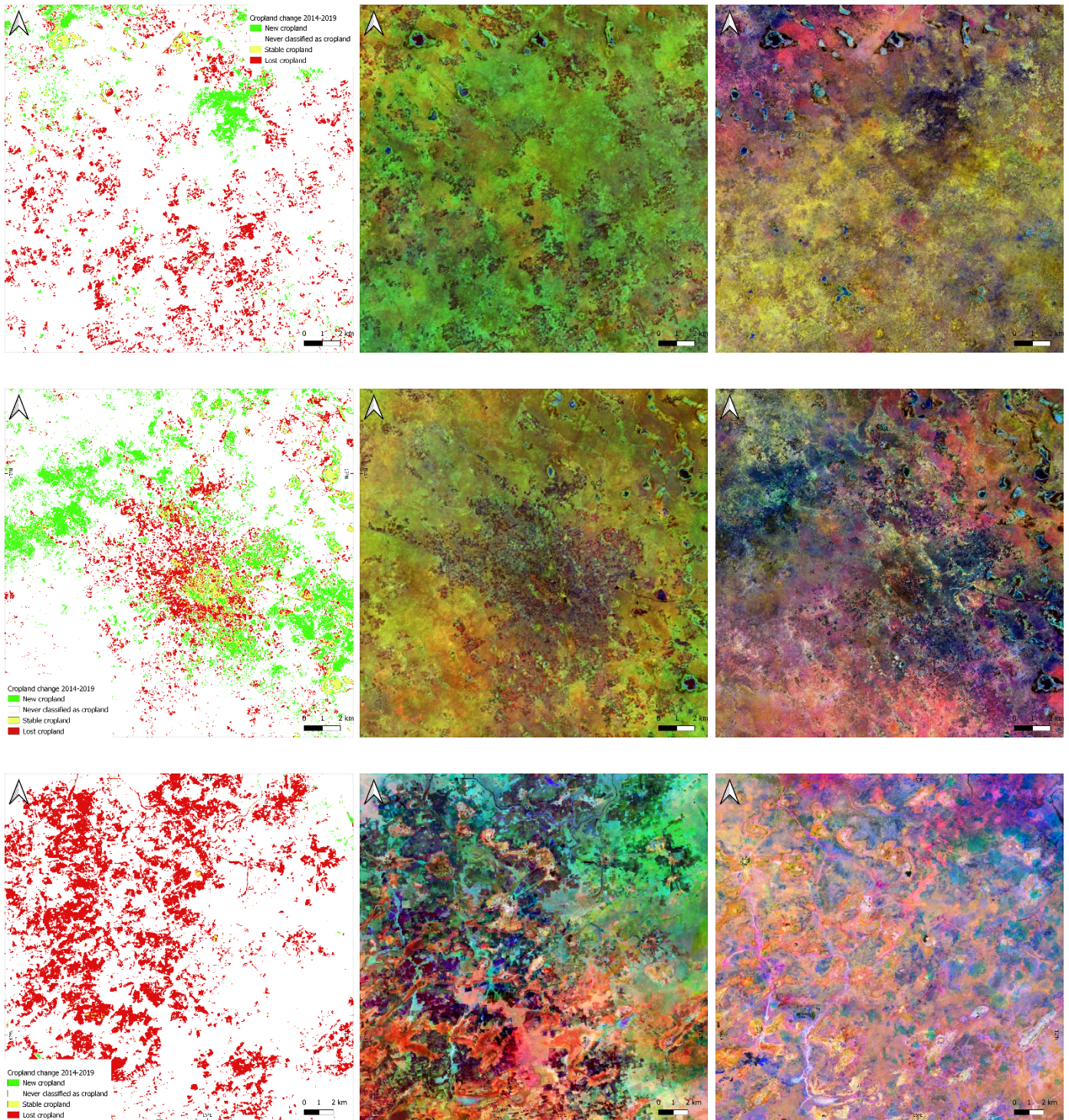
Figure 4.20 Samples close-ups. From left to right: cropland change classification between years 2014 and 2019, 2014 RGB synthesis of per-pixel maximum NDVI composites of August, September and October using Landsat8 imagery, 2019 RGB synthesis of per-pixel maximum NDVI composites of August, September and October using Sentinel-2 imagery. Original satellite imagery in Bottom of Atmosphere (BOA) reflectance values.

.

4.3.   Discussion and conclusions

The result of the classification aligns at least qualitatively with the classification performed visually by Copernicus activation, as illustrated in the following figure. The following map illustrates on top of the cropland change classification obtained with the illustrated methodology, the populated places labelled by Copernicus RRM activation according to the increase, decrease or no change in the surrounding cropland area. There is a significance discrepancy in the two datasets, especially in the Northern area, while in the Kala Balge AOI the two classifications are qualitatively well aligned.
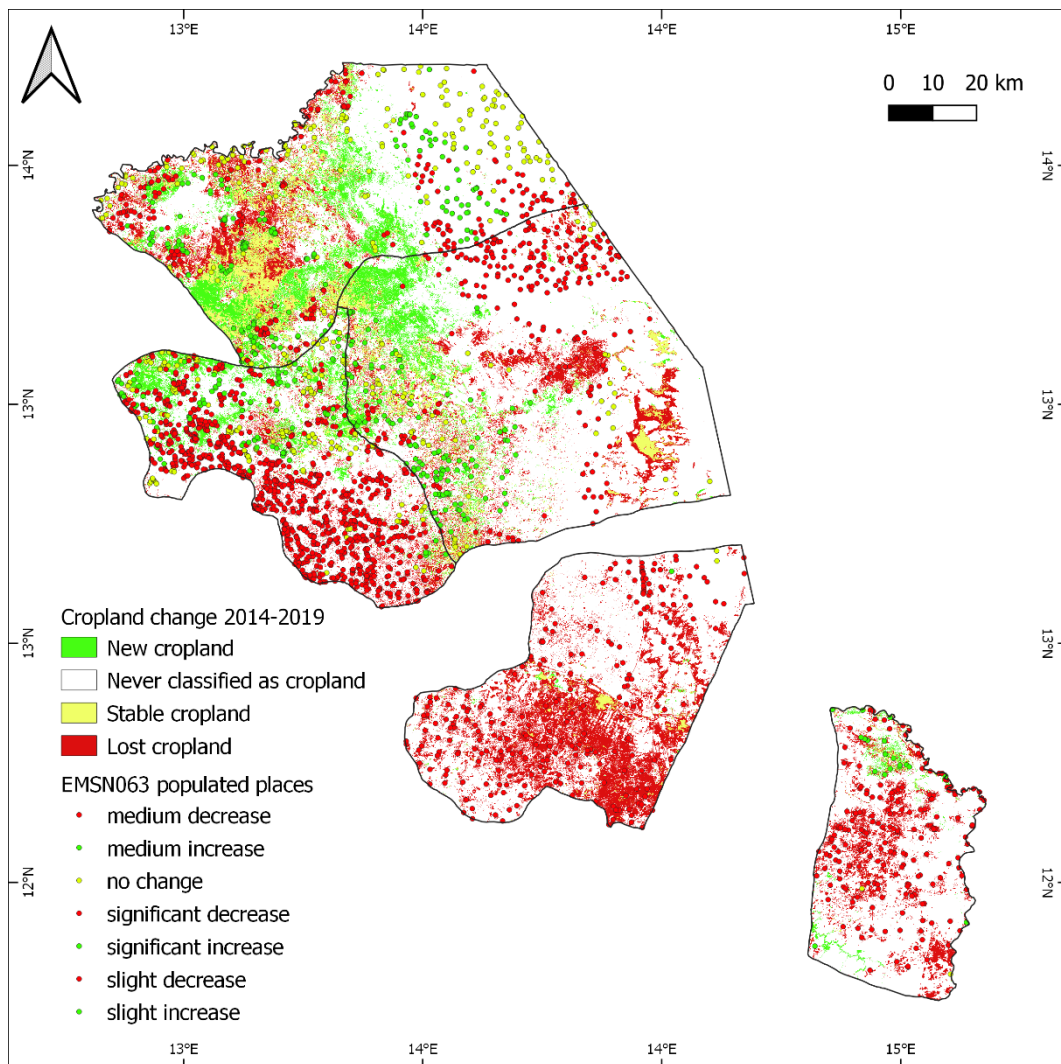
Figure 4.21 Cropland change classification between years 2014 and 2019 map compared to populated places as classified by Copernicus EMSN063 activation according to the change in surrounding cropland status.

Limitations and challenges

The following challenges and limitations were identified within the illustrated approach:

- Pixel-based classification inevitably leads to salt-and-pepper effects in the final products of the classification if not combined with an Object-Based approach.

- The availability of both a training and a validation dataset is highly challenging. In order to perform a supervised pixel-based classification, a training dataset is needed. For the cropland classification, it would be possible to refer to existing coarser resolution cropland masks, such as the 30 meters Global Food Security-support Analysis Data (GFSAD) Cropland Extent generated for reference year 2014-2015 by NASA and USGS (https://lpdaac.usgs.gov/products/gfsad30afcev001/), but their quality was considered not adequate when visually compared to the imagery over the study site. The other challenge is the availability of validation products, in order to compare and evaluate the quality of any obtained classification product when no ground-truth data exist. The visual interpretation and labelling of a validation dataset are a highly challenging and subject to the interpreter task itself.

- The results of the supervised classifications could be potentially improved in terms of validation accuracy metrics by improving the training dataset and the validation datasets by applying an area and class proportional approach when computing the number of training and validation data per each area of interest.

- The impact of clouds albeit the use of maximum NDVI composites generates noise in the classification. The use of a random forest classifier that repeatedly takes shuffled subset of both the training data and the input features minimizing outliers might positively contain the impact of cloud patches in some bands, hence is some monthly per-pixel composites, of the stacks used as training properties/features.

Benefits

The following benefits were identified within the illustrated approach:

- Recurring to high-resolution (up to 10 meters) satellite products allows a finer resolution product compared to already existing yearly cropland or land cover products, such as the Copernicus Proba-v 100 meters yearly (2015-2019) land covers (https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_Landcover_100m_Proba-V-C3_Global).

- The multi-temporal composites approach allows to contain the cloud coverage issue. Several methods exist to control the cloud coverage of imagery in Google Earth Engine. Both the cloud-masking approach

using quality bands (QA60 for Sentinel-2 and BQA for Landsat-8) available in the used satellite products and the quality mosaic approach were investigated to this end. Nevertheless, both approaches, and the mixed approach, lead to imagery artifacts, most probably due to the inaccuracy of the quality bands included in the satellite products. Consequently, the maximum NDVI approach applied over the whole imagery availability (hence without a filter on cloud percentage over the whole image) was applied in order to limit the probability that a cloudy pixel is selected if a sufficient number of images are used as input of a reducer when compositing.

- In order to obtain a supervised classification product only 250 training points had to be generated through a semi-automated process including an unsupervised clustering algorithm and a visual inspection of obtained clusters, compared to more than 1500 points visually inspected in both the WFP and Copernicus activations. This could be a good qualitative input or starting point for focusing further mapping visual efforts.

Chapter 5

5. Crowdsourcing and DIAS

This Chapter, in contrast with the previous ones, is going to explore two additional technologies that mainly focus on a qualitative approach to imagery and on a visual interpretation of it to extract information. Starting with the Volunteered Geographic Information (VGI) of the Humanitarian OpenStreetMap Team and ending with the contribution that new cloud infrastructures could provide to support the efforts of volunteer mappers. All framed in the context of an application for displaced population monitoring.

5.1. Crowdsourcing and the Humanitarian OpenStreetMap Team

In the geospatial world, the "crowd" can be identified as the wide group of volunteers who deliberately and often with no professional background decide to join the cause and, by sitting at their laptops, start looking for vital information on satellite images and producing data that is then translated into essential information. They contribute to the generation of the so called Volunteered Geographic Information (VGI), based on the principles of open and crowdsourced data and participatory mapping.

There is no better way to define OpenStreetMap as the "Wikipedia of maps". OpenStreetMap, from now on referred to with the acronym OSM, is a multi-faceted project that enables distributed work around a common product. OSM is a database that contains geographic data for many parts, nowadays almost all, of the world, it is a website and a set of software and tools that allow users to contribute, download and interact with the database. OSM is also a community that interacts through various channels. People active in OSM participate for a wide variety of reasons but most of them focus on the ideology of and opportunities created by non-proprietary geospatial data (Soden at al, 2014).

OpenStreetMap is the perfect representative of the Volunteered Geographic Information (VGI), representing a free, open and editable map of the world, updated by the minute. Whoever can collect data through, for example, GPS surveys or aerial photography and, at the same time, whoever can digitize the available satellite image creating useful geographic information covering the whole Earth's surface. This crowdsourced information is entered into the database of OpenStreetMap and data can be freely exported from the database into different suitable formats being processed by common GIS software (Harvard Humanitarian Initiative, 2011).

The interest of the OSM community in the humanitarian work started with the 2010 Haiti's earthquake, with a growing contingent who believed that values such as data openness and sharing and civic participation perfectly fit in the emergency and humanitarian response. The Humanitarian OpenStreetMap Team, from now on usually referred to with the acronym HOT, was firstly created in 2010 to build the capacity of OSM contributors and volunteers and to draft a strategy for how to best support the relief efforts (Soden et al., 2014). HOT/OSM became a global community of volunteers that are asked to rapidly digitize available satellite imagery whenever a humanitarian crisis occurs or an humanitarian task is asked to be solved, obtaining as a final result maps and data which may support humanitarian organizations in delivering their relief aid or prevention plans. Most of HOT's response activities occur remotely. After a disaster strikes, HOT members search for existing data and available satellite imagery. Pertinent partners are contacted to provide compatible imagery. Once the imagery is obtained, the virtual community digitizes, or traces from the imagery. Normally, the focus is on recognizable objects that are useful for humanitarian response, like roads, buildings, blocked roads, flood extent, and generates data and maps. During this time, responding organizations are also contacted to determine their needs (LearnOSM, 2015). In between the on the field humanitarians and the OpenStreeMap community, the Humanitarian OpenStreetMap Team takes the role of coordinating the two groups in order to concentrate the construction of data where it is more needed, for instance by contacting the pertinent organizations to determine their most urgent needs. HOT nowadays is a non-profit organization coordinating simultaneously the remote community and the field community with on-going capacity building (HOT, 2019).
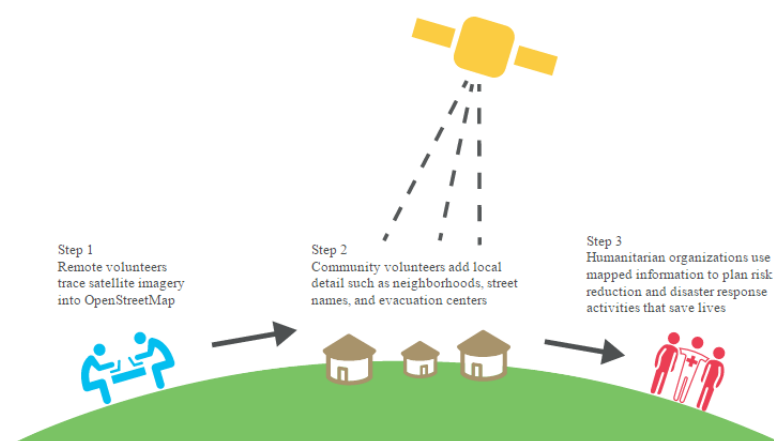
Figure 5.1 Workflow of a typical HOT/OSM activation.

5.1.1. The Missing Millions project

The Missing Millions project is an initiative funded by NASA and led by Oregon State University in collaboration with the Humanitarian OpenStreetMap Team and Development Seed. The main goal of this project is the creation of an open global database of informal settlements location and schema in order to monitor Sustainable Development Goals within them (SDGs). To achieve this, remote sensing, crowdsourcing and machine learning are planned to be leveraged. The steps of the project are the following:

- Collection of global data on already known informal settlements;
- Training of a machine learning algorithm based on known informal settlements;
- Location of as-yet-unrecorded informal settlements through a crowdsourcing campaign;
- Location and characterisation of global informal settlements with machine learning;
- Measurement and monitoring of SDGs in informal settlements (HOT, 2017).

One of the first steps of this project is identified as the location of unknown informal settlements through the OpenStreetMap crowdsourcing community recurring to the MapSwipe application. MapSwipe is an application developed in the context of the Missing Maps Project mainly by the HeiGIT (Heidelberg University GIScience Research Group) to create an easy mean for the crowd to contribute to humanitarian mapping (OpenStreetMap Wiki, 2019).

5.1.2. MapSwipe

The operating principle of MapSwipe sees the users swiping through satellite images of a region and tapping the screen when they identify the features they are asked to look for (buildings, settlement, roads, etc.). The information retrieved through MapSwipe is then used to narrow down the areas fed to mappers for detailed mapping, so that they do not have to scroll a huge number of images over inhabited areas before they actually encounter the features of interest (OpenStreetMap Wiki, 2019).

In the case of the Missing Millions project, MapSwipe is planned to be deployed over large-scale areas to identify potential informal settlements. A novel approach is to use before and after satellite images pairs for change detection. The change detection technique is supposed to support the crowd to better identify informal settlements that have established in between the two periods. The output from the Missing Millions MapSwipe campaign is going to be used to guide detailed informal settlements mapping and the results of the detailed mapping are going to feed a machine learning algorithm.

5.1.3. A potential case study for the ONDA DIAS application

The Missing Millions project and the deployment of a MapSwipe campaign were seen as a proper opportunity to test the capacity of newly available cloud infrastructures such as the Data Information and Access Services (DIAS), which are going to be further explored in the next sections. It was decided to generate potential test inputs of a MapSwipe campaign in terms of satellite imagery recurring to this newly available infrastructure. The following sections are going to explore the methodology and tools followed and explored to respond to this potential task. The actual imagery fed into the actual MapSwipe campaign that is going to be launched were instead generated within a research group at the Oregon State University.

The main objective of this potential case study application of the ONDA DIAS is identifying any problem in the usability of the infrastructure itself and the generation of a reusable pipeline. In addition, it is of interest the evaluation of the imagery used and its suitability for settlements detection by a community of volunteer mappers.

The test area identified is the Northern Uganda region. The Northern Uganda region is indeed home of refugees in nineteen well-established refugees' settlements in the districts of Adjumani, Arua, Moyo, Lamwo, Koboko and Yumbe.

It was decided to recur to Sentinel-2 imagery, firstly because it has already been proved how this kind of imagery offers the possibility to visually detect settlements in this area (Chapter 2). In addition, the 10 meters spatial resolution of this kind of data is also a topic of research in the development of the app.

Two mosaics of Sentinel-2 imagery are required in a potential MapSwipe campaign for providing a pre- and post-event example to the users, hence the Sentinel-2 images had to be collected for two separate time periods in order to allow before and after change detection. The following requirements were identified for the mosaics:

- The two time periods had to be at least two years apart in order to capture new settlements establishments and allow the crowd to better identify the features by comparing with an image where the feature was not yet present;
- The two time periods had to be selected from the same season or in similar phenological conditions in order to minimize environmental discrepancies between before and after images and ease the work of the crowd;
- The output mosaics should have a good quality in terms of cloud coverage in order not to excessively limit the work of the crowd.

5.2. Tools and methods

This section is going to present a theoretical background of the technical aspects involved in the generation of a Tile Map Service and of the tools and methods leveraged in the generation of such services.

5.2.1. Tiling and Tile Map Services

MapSwipe uses satellite imagery provided by Tile Map Services (TMS) (MapSwipe Back-End, 2019). Consequently, a full pipeline to deliver the two Sentinel-2 mosaics in TMS format had to be developed.

Tiled web maps, slippy maps (as referred to in OpenStreetMap) or tile maps (both raster or vector) are maps displayed within the browser by means of several image or vector data files that are individually requested through the internet and seamlessly joined. A tiled web map is the most common way to display and navigate a map on the internet, while a WMS (Web Map Service) typically displays the single large image allowing to pan nearby areas in the map.

A tiled map is actually constituted by a large number of smaller images, named tiles, which are drawn next to each other in order to build a larger seamless image. The result is one apparent large map image.

Figure 5.2 Example of tiling (Liedman, 2020).

The amount of detail of a map is limited by its size, meaning that a higher level of detail will require either a higher image size or a reduced map scale, namely a higher zoom level. This problem is solved in tiling by pre-originating and storing several maps of the same geographic area at different scales, zoom levels, from the larger to the smaller possible. In order to avoid handling extremely large files when displaying an image at the maximum zoom level, in map tiling, only the images covering the area currently being displayed are required. This is achieved by pre-dividing each image at each zoom level into several smaller images, namely the tiles, of a fixed size. When displaying the image, the viewing application (e.g. the

web browser) will request an image at the current geographic location and the server will provide it.

In conclusion, a tiled map consists of several zoom levels. Each zoom level is a map of the same geographic area drawn at different scales and split into tiles and each tile has the same size independently from the zoom level. Since increasing the scale of the same map can be translated in increasing the map's size in pixels, increasing the scale also increases the number of tiles of the zoom level. Each time a user of the web map pans the map, only the relevant tiles are displayed while new tiles are fetched. This mechanism allows individual tiles to be pre-computed so that the images have not to be rendered in the browser, a quite computationally demanding task.

Tiled web maps belong in the wider framework of client/server mapping solutions. Consequently, the request of tiled web maps by clients and the provision of tiled web maps by the servers has to be well understood.

Of course, conventions or standards exist to address tiled web maps and the client/server relationships. The main properties of tiled web maps, namely the size of the tiles, the numbering of the zoom levels, the projection to use, the numbering and identification of individual tiles and the method to request them are standardized by different existing conventions. Indeed, tiles have to be numbered according to numbering schemes and they are usually managed into rows and columns (up/down the firsts and left/right the latter). This rows/columns schema is used to identify the tile of belonging of each pixel constituting the map or image. This is done by subtracting from the pixel coordinate the tile set's origin and dividing the result by tile size in pixels.

The request and display of a tiles map in the browser, instead, requires the support of a web mapping framework to handle the retrieval of tiles, the display, the caching and the user experience itself. Very popular web mapping framework are OpenLayers or Leaflet.

The most common conventions for standards existing for tile web maps are TMS (Tile Map Service), the XYZ OpenStreetMap standard or Slippy Map Tilenames, WMTS (Web map Tile Service) that is the most recent and more complex OGC (Open Geospatial Consortium Standard), etc. (Wikipedia, 2019; Liedman, 2020).

Main standards common across conventions are:

- Tiles are 256x256 pixels;
- The outer most zoom level is 0 and the entire world is rendered in a single map tile at this zoom level;

- At each zoom level both dimensions are doubled. This means that a single tile is replaced by 4 tiles when zooming in;
- The projection used for tile web maps is Web Mercator EPSG:3857;
- The images are served through a web server by means of an URL. In general, all tile web maps expect tiles to be served up by URLs following their specific standard's scheme.

The OpenStreetmap standard for the Slippy Map Tilenames known also as XYZ adds the following conventions:

- The number scheme is X and Y, where X and Y identify the single tile;
- The tiles are PNG format files;
- Images are server through a web server by means of an URL in the form of http://.../Z/X/Y.png where Z is the zoom level, X and Y identify the tile;
- Each zoom level is a directory, each column is a subdirectory and each tile in that column is a file. This convention is followed by TMS as well.
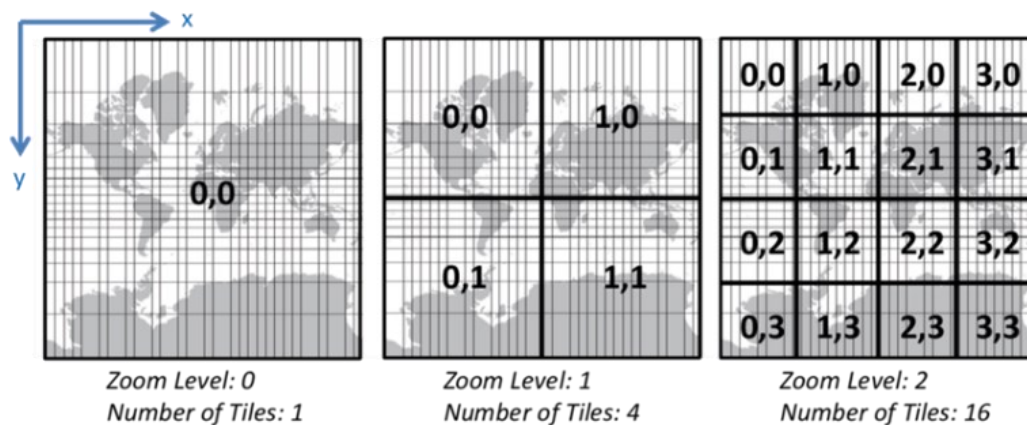
Figure 5.3 Numbering scheme in XYZ standard (Stefanakis, 2017).

The OpenStreetMap XYZ standard has a numbering scheme that goes (0 to 2zoom-1, 0 to 2zoom-1) for the range (-180, +85.0511) - (+180, -85.0511). The specific numbering scheme for the Tile Web Map (TMS) standard instead is (0 to 2zoom-1, 2zoom-1 to 0) for the range (-180, +85.0511) - (+180, -85.0511). This means that while for XYZ standard the x and y are growing right and downwards starting from top left, for the TMS standard the x and y are growing right and upwards starting from bottom left. Basically, the y is inverted for TMS and XYZ. The zoom parameter is an integer between 0 and a maximum of usually 18.
Each zoom level has a different resolution, measured in meters per-pixel. Both a formula and reference tables for calculating the resolution at each zoom level exist.

In general, all tile web maps expect tiles to be served up by URLs following their specific standard's scheme. The first part of the URL usually indicates the server where the tiles are hosted (Wikipedia, 2020a).

5.2.2. Data Information and Access Services (DIAS)

To work on generating TMS to be potentially deployed in the MapSwipe app, it was necessary to retrieve the required satellite imagery, to process the imagery and to make it available on a server to be published on the browser or in the app to the users. Since it was planned to recur to Sentinel- 2 data, it was decided to recur to a new technology in the field of satellite imagery, the Data Information Access Services (DIAS).

DIAS are cloud-based infrastructures developed in the year 2018 in the framework of a European Commission (EC) initiative. The main objective of this initiative was to facilitate the users' access to Copernicus data and information services. DIAS provide a single centralized access to the full set of Copernicus data, including Sentinel data (more than 10 petabytes delivered each year) and information products from the six operational services. In addition, DIAS offer the possibility and capacity to manipulate, process and download data, mostly with open source tools (Copernicus, 2017; Copernicus, 2020a).

The Data and Information Access Services provide the users the great possibility to develop and deploy their own applications in the cloud. In addition, DIAS provide the innovative advantage of bypassing the heavy download and storage of bulky dataset locally. Indeed, the capacity required for managing the massive amount of Copernicus data has always represented a complex challenge in the past (Copernicus, 2017; Copernicus, 2020a). DIAS can be seen as the European answer to the well-known Google Earth Engine. Different fields of application are in the reach of DIAS: climate change, land monitoring, marine environment, atmosphere monitoring, security and emergency and disaster management (UN-SPIDER, 2018).

Five different DIAS have been funded to be developed, all based on a cloud-based system architecture: CREODIAS, MUNDI, ONDA, SOBLOO and WEKEO (Copernicus, 2017; Copernicus, 2020a).
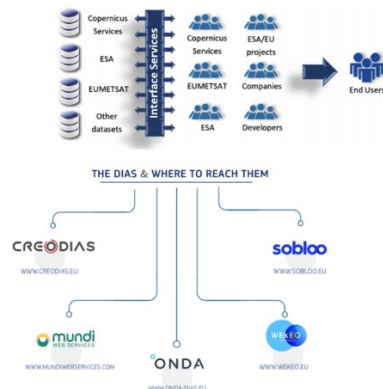
Figure 5.4 DIAS dissemination workflow (Copernicus, 2020a).

In the context of this research, it was possible to get access to the ONDA DIAS infrastructure. ONDA gives the possibility to deploy a virtual server designed according to the user's needs. The virtual server is already provided with useful tools for geospatial processing such as python and the *gdal* library and it was further enriched with processors such as *Sen2Cor* for processing of Sentinel-2 imagery.

The ONDA DIAS Advanced Application Programming Interface (API) is based on the Elastic Node Server (ESN), a specific system allowing the access to low level components of the satellite products without a full download (ONDA; 2018). The ENS is a Linux-based Open Source software developed by GAEL Systems (GAEL, 2017). ENS basically provides a simple and scalable front-end to one or more Data Storages. The ESN is mounted on the Computing Instances of ONDA DIAS and allows the Data Storages to be accessed and used with the specific client Computing Instances. A Computing Instance is the server running the applications of the specific user, namely a virtual machine running the application developed by the user in the cloud. ENS acts like an interface between the Computing Instance and the complexity of Files opening, Files format discovery and data access engineering skills. ENS exposes only the resulting unified tree of Nodes through standards and well-defined Distribution File Systems such as the Network File System (NFS). A DFS is a methodology to store and access files based on a client/server architecture, where the files are stored by one or more central servers and remote clients in a network can access them. ENS adds to conventional File and Object Storages the organization of Directories and Files into further logical trees of Nodes. These Nodes can be located, queried and accessed semantically by means of names despite their physical formats and their locations (ONDA, 2018). ENS can in addition handle a variety of data types. A central and unified data model is used, allowing a separation between the data management logic and the implementation details peculiar of each data format and access method. ENS Data Model supports already a variety of EO products, such as Sentinel-1/2/3, ENVISAT ASAR, Landsat-8, etc. (ONDA, 2018). The default ENS is mounted on the ONDA DIAS Virtual Server. All the products available in the ONDA Catalogue are stored in the ONDA ENS, even though some of them may require to be ordered.

Since the ONDA DIAS archive contains millions of products, these are spread into directories hierarchy following a "What/When/Where" organisation. The first set of directories and sub-directories concert what kind of data is exposed, the second set of sub-directories when the data is acquired and the last set concerns where the data is located. The information used to organise the ENS directory tree is extracted by the metadata of the products when they are ingested. The result of this directories organization is a series of "pseudo-paths" per product (ONDA, 2018).

Access to the products archived in the ONDA DIAS catalogue can be performed, among others, in two main ways:

-   Through the ONDA DIAS online catalogue.
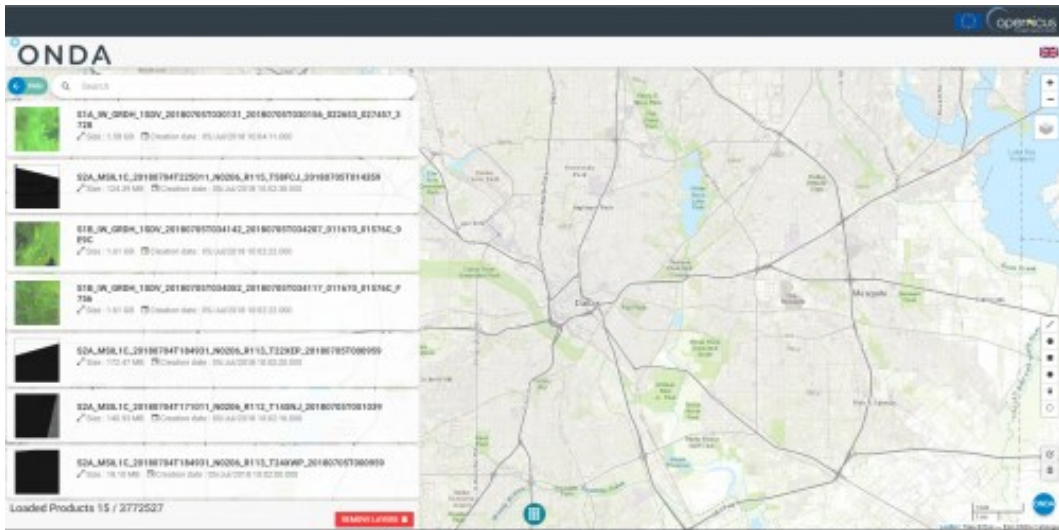
Figure 5.5 The ONDA online catalogue interface.

The online ONDA catalogue allows to browse the products per area, sensing period, product type and cloud coverage. Once the products are identified they can be downloaded in .zip format or ordered whether they are archived. Usually, products older than three months in sensing date are archived. The online ONDA catalogue allows also to browse the specific products' information and details.
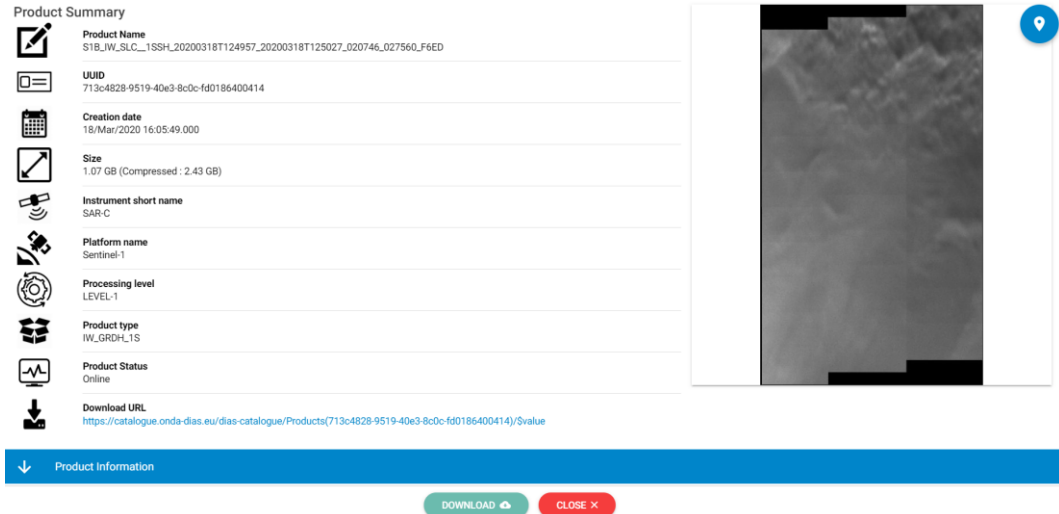
Figure 5.6 Product details available in the ONDA online catalogue.

-   Through the ONDA architecture itself, using the Advanced API service base on the already explained ENS, without having to download the product.

An important information of the products, made available when investigating the products online, is the UUID (Universally Unique Identifier). The UUID allows to send request to the server via the Open Data Protocol and to have in return a response, usually in XML format. The XML response, if no other specific query is

performed, includes relevant products metadata including the product name and the product "pseudo-paths" in the ONDA infrastructure, as illustrated in the image below (ONDA, 2018).

Figure 5.7 XML response to a ONDA server query.

All of this is possible via an Open Data Protocol (OData) interface exposed by the ONDA Catalogue. The OData is an open protocol or a set of standards allowing the standardises generation and use of query-able RESTful APIs. Basically, yhr OData defines conventions, rules, and formats for data handling on the web using HTTP requests (Hypertext Transfer Protocol). The OData standards define the logic for constructing the URIs identifying data and metadata exposed by an OData server and URI query strings operator to query the same data and metadata. Through the Representation State Transfer architecture (REST), resources are identified using the Uniform Resource Identifier (URI), they are published and edited by Web clients using HTTP messages (ONDA, 2018).

5.2.3.   Pipeline

After identifying the tools available for the development of a TMS, it was possible to delineate the suitable pipeline for the work. The pipeline consisted in the following steps.

Selection of the pre- and post-periods

This step comprises the selection of the two periods for the acquisition of the satellite imagery. As already stated before, it was necessary to identify two periods apart of at least two years and that showed similar phenological activity across the area of interest (AOI). The similarity in phenological activity is not related to any calculation of vegetation indices, but it is strictly related to the expected enhanced capability of the crowd to detect better the features (the informal settlements) over two pre- and post-images, in which environmental differences are minimized as much as possible.

The post period was identified in the two months of January and February 2020. It was indeed decided to use the most recent available imagery for the post mosaic. In the following image, an overview mosaic of the least cloudy (0.1 %) Sentinel-2 imagery covering the AOI in January and February 2020 was retrieved using Google Earth Engine.
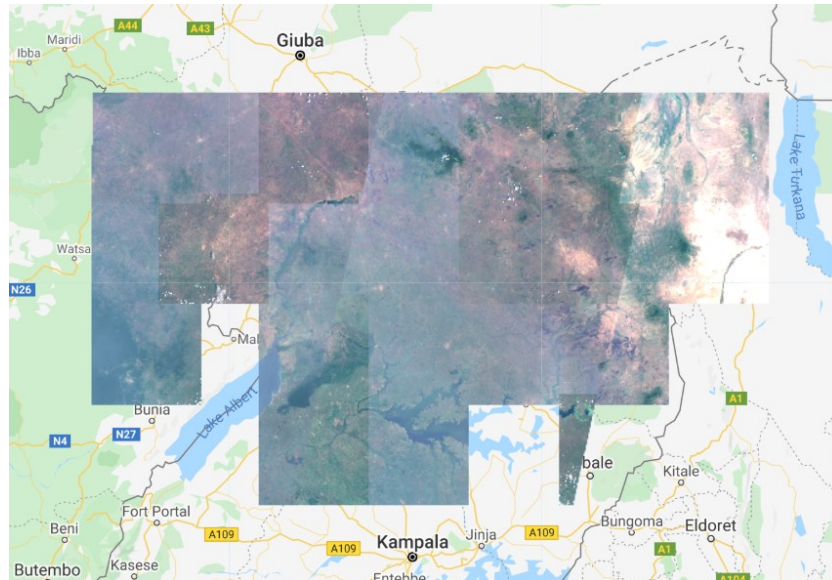
Figure 5.8 Mosaic over the AOI for the period January and February 2020 in Google Earth Engine.

Another overview mosaic similar to the previous one was generated exactly two years backwards, for January and February 2018. From the image below it is evident how the 2018 mosaic does not show similar phenological activity as the 2020 mosaic, since 2018 shows a quite drier vegetation compared to 2020.

Figure 5.9 Mosaic over the AOI for the period January and February 2018 in Google Earth Engine.

It was decided then to move the selection of the images backwards. The overview result for December 2017 shows a phonologically aspect more similar to

2020, at least from a visual inspection. Hence it was decided to select December 2017 as a pre period for the generation of the mosaics.
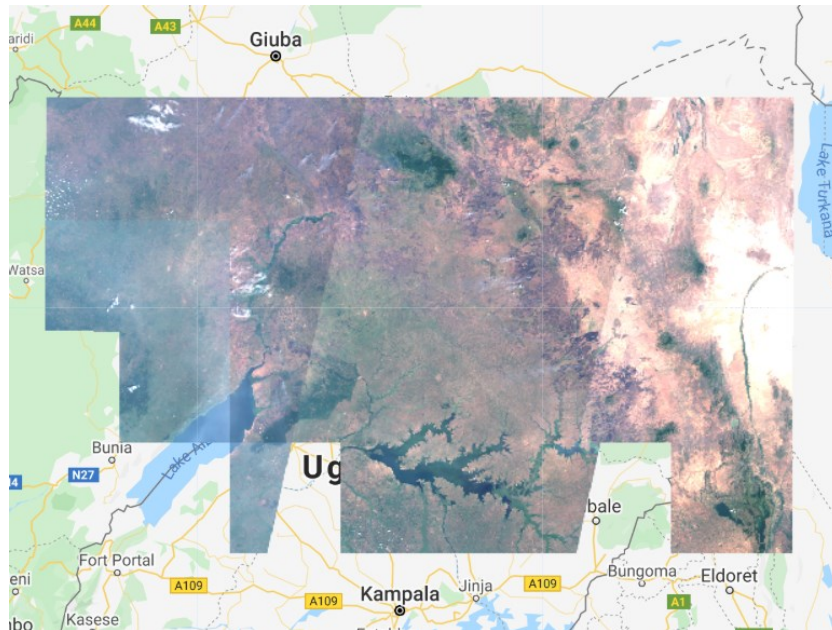
Figure 5.10 Mosaic over the AOI for the period December 2017 in Google Earth Engine.

Selection of the imagery

Once the two periods were identified, the single scenes of Sentinel-2 had to be identified and selected. The AOI was intersected with the Sentinel-2 tiles grid.
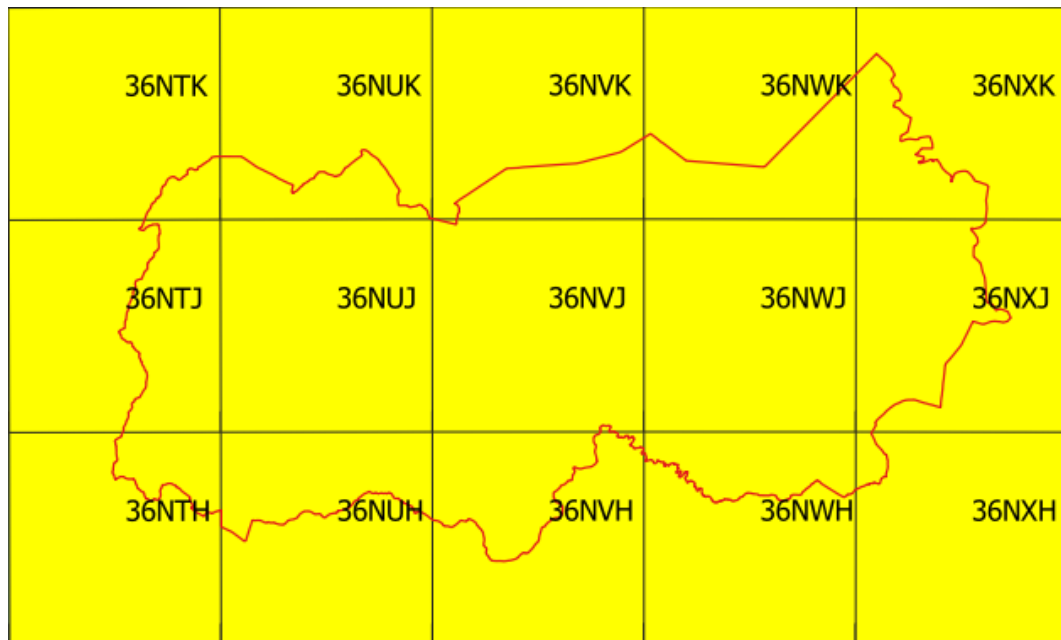
Figure 5.11 Sentinel-2 tiles covering the AOI.

Some of the tiles are constituted by two different acquisitions because of the acquisition trajectory. This is better explained in the image below.
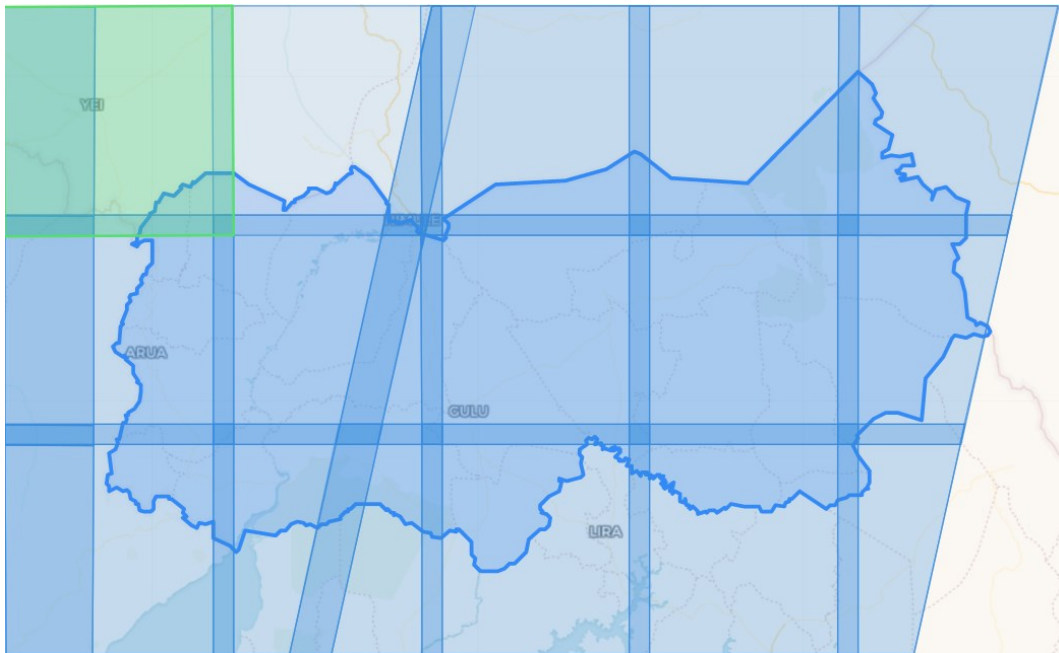
Figure 5.12 Sentinel-2 tiles covering the AOI with effect of the sensor swath.

Once the Sentinel-2 tiles intersected by the AOI were identified, the selection of optimal Sentinel-2 scenes was performed directly in the online ONDA catalogue. By filtering the results by AOI, dates of interest corresponding to the two identified periods and by cloud coverage (a limit of 0.1% was used), a list of possible Sentinel-2 scenes was returned. Among this, given the limited number of tiles covering the AOI, it was possible to visually inspect and select the best imagery to be processed in the developed pipeline.

Following the methodology explained in the DIAS section, the "pseudo-paths" of the selected Sentinel-2 were identified and stored in a .txt file to be then implemented in the automatic procedures. The Sentinel-2 scenes selected for the post period are:

```
S2A_MSIL2A_20200212T081041_N0214_R078_T36NTH_20200212T113212
S2A_MSIL2A_20200212T081041_N0214_R078_T36NTJ_20200212T113212
S2A_MSIL2A_20200212T081041_N0214_R078_T36NTK_20200212T113212
S2A_MSIL2A_20200212T081041_N0214_R078_T36NUH_20200212T113212
S2A_MSIL2A_20200209T080051_N0214_R035_T36NUH_20200209T110919
S2A_MSIL2A_20200212T081041_N0214_R078_T36NUJ_20200212T113212
S2A_MSIL2A_20200209T080051_N0214_R035_T36NUJ_20200209T110919
S2A_MSIL2A_20200212T081041_N0214_R078_T36NUK_20200212T113212
S2A_MSIL2A_20200209T080051_N0214_R035_T36NUK_20200209T110919
S2A_MSIL2A_20200209T080051_N0214_R035_T36NVH_20200209T110919
S2A_MSIL2A_20200209T080051_N0214_R035_T36NVJ_20200209T110919
S2A_MSIL2A_20200212T081041_N0214_R078_T36NVK_20200212T113212
S2A_MSIL2A_20200209T080051_N0214_R035_T36NVK_20200209T110919
S2A_MSIL2A_20200120T080241_N0213_R035_T36NWH_20200120T111107
S2A_MSIL2A_20200209T080051_N0214_R035_T36NWJ_20200209T110919
S2A_MSIL2A_20200209T080051_N0214_R035_T36NWK_20200209T110919
S2A_MSIL2A_20200209T080051_N0214_R035_T36NXH_20200209T110919
S2A_MSIL2A_20200209T080051_N0214_R035_T36NXJ_20200209T110919
S2A_MSIL2A_20200117T075241_N0213_R135_T36NXJ_20200117T105226
S2A_MSIL2A_20200209T080051_N0214_R035_T36NXK_20200209T110919
```

The Sentinel-2 scenes selected for the pre period are:

```
S2A_MSIL1C_20171214T081331_N0206_R078_T36NTH_20171214T103532
S2A_MSIL1C_20171214T081331_N0206_R078_T36NTJ_20171214T103532
S2A_MSIL1C_20171214T081331_N0206_R078_T36NTK_20171214T103532
S2A_MSIL1C_20171214T081331_N0206_R078_T36NUH_20171214T103532
S2A_MSIL1C_20171221T080331_N0206_R035_T36NUH_20171221T100353
S2A_MSIL1C_20171214T081331_N0206_R078_T36NUJ_20171214T103532
S2B_MSIL1C_20171206T080309_N0206_R035_T36NUJ_20171206T102614
S2A_MSIL1C_20171214T081331_N0206_R078_T36NUK_20171214T103532
S2B_MSIL1C_20171206T080309_N0206_R035_T36NUK_20171206T102614
S2B_MSIL1C_20171206T080309_N0206_R035_T36NVH_20171206T102614
S2B_MSIL1C_20171216T080319_N0206_R035_T36NVJ_20171216T102645
S2B_MSIL1C_20171219T081329_N0206_R078_T36NVK_20171219T105309
S2A_MSIL1C_20171221T080331_N0206_R035_T36NVK_20171221T100353
S2B_MSIL1C_20171206T080309_N0206_R035_T36NWH_20171206T102614
S2B_MSIL1C_20171206T080309_N0206_R035_T36NWJ_20171206T102614
S2A_MSIL1C_20171221T080331_N0206_R035_T36NWK_20171221T100353
S2B_MSIL1C_20171206T080309_N0206_R035_T36NXH_20171206T102614
S2B_MSIL1C_20171206T080309_N0206_R035_T36NXJ_20171206T102614
S2B_MSIL1C_20171213T075319_N0206_R135_T36NXJ_20171213T101202
S2B_MSIL1C_20171216T080319_N0206_R035_T36NXK_20171216T102645
```

As it can be seen, the pre period Sentinel-2 imagery belong to the Sentinel-2 Top of the Atmosphere (TOA) reflectance products Level-1C (L1C). This means that, in order to eliminate the effect of atmosphere on these scenes, the atmospheric correction must be performed to obtain Bottom of the Atmosphere (BOA) reflectance products Level-L2A (L2A). Indeed, Level-2A products were systematically generated by the European Space Agency (ESA) ground segment over Europe since March 2018 and extended globally in December 2018. All the previous products are available in L1C format and tools exist for the generation off the associated L2A product, such as the Sen2Cor processor.

All products available in the ONDA DIAS infrastructure are in the original SAFE format. In the SAFE format, both for L1C and L2A product, the tiles files with TOA/BOA reflectance image data file are stored as GML/JPEG2000 for each band.
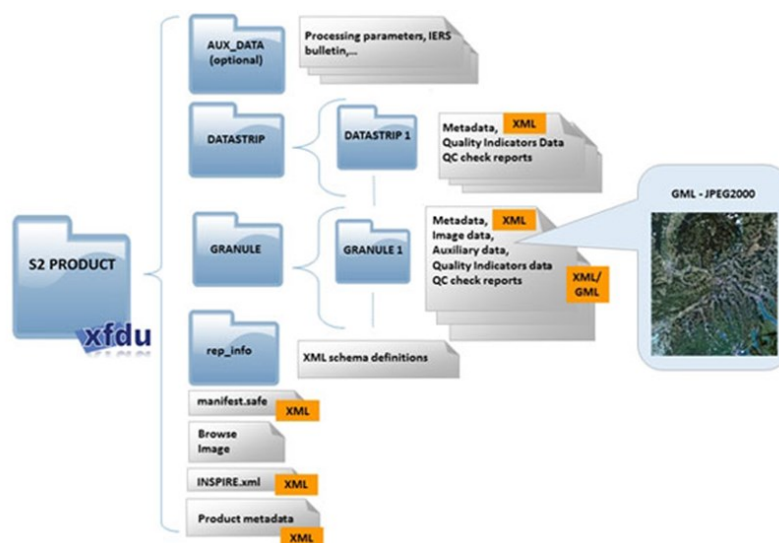
Figure 5.13 Sentinel-2 data product structure.

Sen2Cor, i.e. Sentinel 2 (atmospheric) Correction, is a processor for Sentinel-2 Level 2A product generation and formatting provided by the European Space Agency. *Sen2Cor* performs a pre-processing of L1C Top of Atmosphere image data, performing the atmospheric, terrain and cirrus correction of TOA L1C input data. The processor creates an ortho-image Level-2A Bottom-Of-Atmosphere reflectance product. Outputs are an Aerosol Optical Thickness (AOT) map, a Water Vapour (WV) map and a Scene Classification map together with Quality Indicators for cloud and snow probabilities. Its output product format is equivalent to the Level 1C User Product: JPEG 2000 images, three different resolutions, 60, 20 and 10 m according to the native resolution of the different spectral bands.

Input of the *Sen2Cor* processor are the Level 1C User Product Top of Atmosphere image data, consisting in 100 km square tiles. Each tile consists of thirteen compressed JPEG-2000 images, each image representing one single band. The thirteen bands have three different resolutions (10m, 20m and 60m).

Output of the *Sen2Cor* processor are Level-2A Product Bottom-Of-Atmosphere (BOA) reflectance product. The Level-2A image data product uses the same tiling, encoding and filing structure as Level-1C. Each L2A product folder consists of a series of sub-folders and the sub-folder named "GRANULE" includes the tile constituting the Level-2A user product. The "QI_DATA" folder includes the Scene Classification and the Quality Indicators for snow and cloud probability at 20 m and 60 m. The "IMG_DATA" folder includes BOA raster at three different spatial resolution and scene classification raster at 20 and 60 m of spatial resolution.

Figure 5.14 Effect of the application of the Sen2Cor processor over a Seninel-2 L1C product (left) generating a Sentinel-2 L2A product (right).

Pre-processing and tiles generation

Once the Sentinel-2 scenes are identified, they cannot simply be used to generate the TMS but they need to be pre-processed in order to be suitable for tiling and web publishing. A completely automated workflow was the ultimate goal. In order to do so, a Python 2.7 script was built based on the *os*, *subprocess* and *gdal* libraries.

Gdal is a free and open source library for raster and vector geospatial data formats. *Gdal* was released under Open Source Licence by the Open Source Geospatial Foundation. It supports many formats of both vector and raster data with a single abstract data model for each and it also provides a variety of command line utilities for data translation and processing (GDAL, 2019).

The *gdal* utility used for the tiles' generation is *gdal2tiles.py.* This utility allows to create a directory with the tiles and the respective metadata following the OSGeo Tile Map Service Specification. In addition, the utility also creates simple web pages with viewers based on Google Maps, OpenLayers and Leaflet. One of the prerequisites for *gdal2tiles.py* to work properly is that inputs should be Byte data types not to cause wrong outputs when converting non-Byte data type to Byte. In addition, as explored in the TMS section, one of the prerequisites of a TMS is that the projection used for tile web maps should be Web Mercator EPSG:3857. In conclusion, two of the main prerequisites to be guaranteed by the automated procedure are the Byte data type and the EPGS:3857 projection of the input imagery.

The main steps of the automated procedures where then identified as:

- Search of the useful bands in the SAFE format of the Sentinel-2 scenes. A true colour composite wants to be created at first for the MapSwipe campaign, hence the Red, Blue and Green .jp2 files representing the bands of the Sentinel-2 images (B02, B03, B04) at 10 meters resolution have to be automatically identified and selected;

- Atmospheric correction and generation of L2A product if the product processed is a L1C with the *Sen2Cor* processor. Sen2Cor (both versions) was previously installed on the ONDA DIAS server;

- Bands composite with the *gdal_merge.py* utility. Once the useful .jp2 files are selected, they have to be composited into a GeoTIFF files, each representing one Sentinel-2 scene in true colours;

- Reprojection of the bands composite GeoTIFF files into EPSG:3857 projection with the *gdalwarp.py* utility;

- Translation with the *gdal_translate.py* utility of the GeoTIFF files from Sixteen-bit unsigned integer to Byte data type. This procedure also requires rescaling the reflectance values of the GeoTIFF files from their actual scale of values for each band to a 0-255 scale in a Byte data type. The actual rescale scale was set to 1-255 in order to avoid the generation of misleading no-data where reflectance values were originally low in the images. This translation procedure allows to set the final appearance of the single Sentinel-2 scenes before the tiling and web publishing, acting as an

histogram stretching procedure, cutting the reflectance values to a minimum and maximum values to improve the appearance of the images;

- Construction of a virtual dataset mosaic of the GeoTIFF files with the *gdalbuildvrt* utility. Virtual datasets are very handy when large datasets constituted by multiple files have to be managed. A virtual dataset is essentially an XML file describing to gdal where the actual files composing the virtual dataset are located. Virtual datasets, consequently, allow handling large dataset as if they were single files. A gdal Virtual Format (VRT) is a format driver for gdal allowing to compose a virtual dataset from other gdal datasets. VRT description in XML are saved with a .vrt e extension. The advantage of a gdal VRT is that, such as in the case of a mosaic constituted by several single GeoTIFF files, a virtual mosaic is much lighter and manageable than a full GeoTIFF mosaic (CSC, 2020);

- Tiling. As a final step before the publishing of the tiles through a web server, tiles are generated using the *gdal2tiles.py* utility in gdal. This is the gdal utility allowing to generate a directory containing the tiles following the TMS convention, explained in the dedicated section. The *gdal2tiles.py* automatically generate the tiles starting from the input file, creating a directories structure where each zoom level is a directory, each column is a subdirectory and each tile in that column is a file. The utility allows to set the zoom levels to be generated as tiles. As a good practice, the zoom level is chosen in order to have a maximum resolution (the minimum value of meters represented per-pixel) of the tile equal to the resolution of the imagery used as input. In this case, using Sentinel-2 images, the minimum meters per-pixel than can be represented is 10, and the maximum zoom level should then be 14, where the resolution is 9.5546 m/px.

Publishing

As explained in the tiled web maps section, the images composing the tiled web map are served through a web server with a URL like *http://.../z/x/y.png*, where x and z are the numbering scheme identifying the specific tile and z is the zoom level. The *gdal2tiles.py* utility generates a directory scheme such as the one illustrated in the following figure.
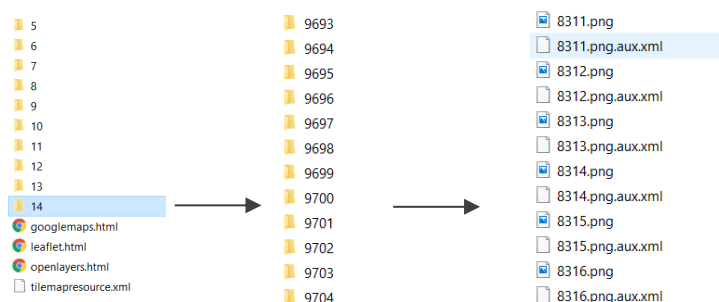
Figure 5.15 Directory scheme generated by the gdal2tiles.py utility.

A web server is a software application running on a server that is able to handle the requests for web pages coming from a client through the HTTP protocol. The web server stores, processes and delivers the resource (such as web pages) to clients. The resource is usually actually stored in the server and the client is usually a web browser that makes requests for a specific resource in the server to the server using HTTP language. The web server is able to map the path component of an URL to the local file system resource of our interest, if we are having a static request, or to an internal program for dynamic request. In the specific case of TMS publishing on the ONDA server, the request is a static request to the tiles resource and the URL specified by the client is relative to the web server's root directory (Wikipedia, 2020b). The URL requested by a client would look something like:

http://dias.ithacaweb.org/missing-millions/uganda_post/tiles/{z}/{x}/{y}.png

The client's web browser would then translate this request into a connection to the server *dias.ithacaweb.org*. While *dias.ithacaweb.org* represents the host, the server, */missing-millions/uganda_post/* represents the GET request, the path to be appended by the web server to the root directory of the web server to retrieve the resource. In a Unix machine such as the one of the ONDA DIAS server, the root directory is usually */var/www*. The result of this request is a link to the local file system resource, in our case the directory with the tiles structure that satisfies the specific TMS's standard's scheme. The web server is able to read the resource and send the proper response to the client's web browser, describing the content of the resource and containing the resource itself (Wikipedia, 2020b).

In this case the web server software is the Nginx web browser installed in the ONDA DIAS infrastructure. The client would potentially be the MapSwipe app. Some simple web map visualizers were also created in order to share and visualize the output TMS. The MapBox GL library was used to this end.

The results are the following TMS, available at the following visualizers:

http://dias.ithacaweb.org/missing-millions/uganda_pre_2017/tiles/{z}/{x}/{y}.png
http://dias.ithacaweb.org/missing-millions/uganda_post/tiles/{z}/{x}/{y}.png

http://dias.ithacaweb.org/missing-millions/uganda_post/
http://dias.ithacaweb.org/missing-millions/uganda_pre_2017/

Update: in February 2021, the server hosting the ONDA DIAS underwent a failure leading to the loss of the generated TMS previously reachable by the reported URLs.

In the next images the TMS are visualized imported into QGIS.
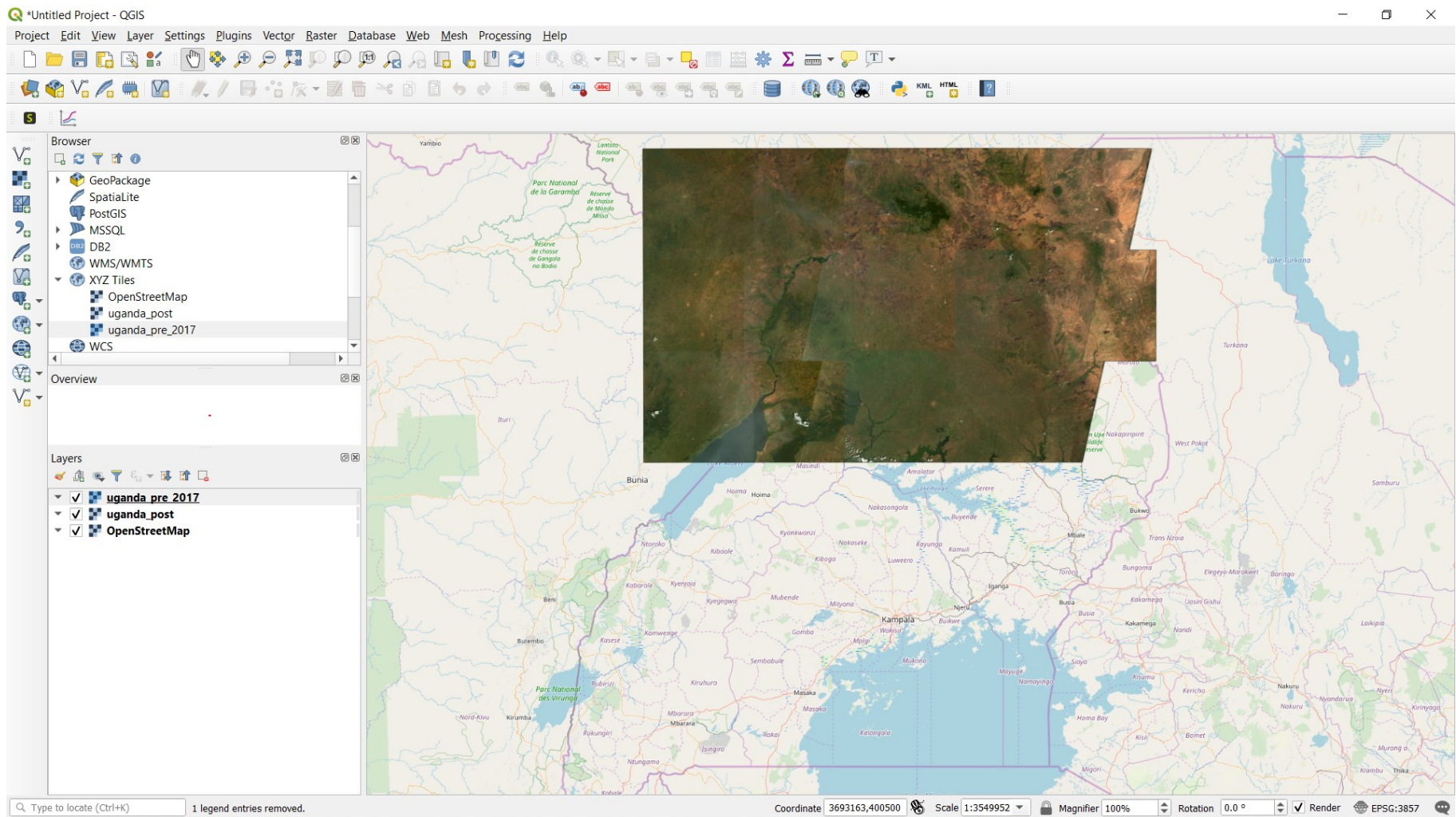
Figure 5.16 TMS for 2017 imported into QGIS.
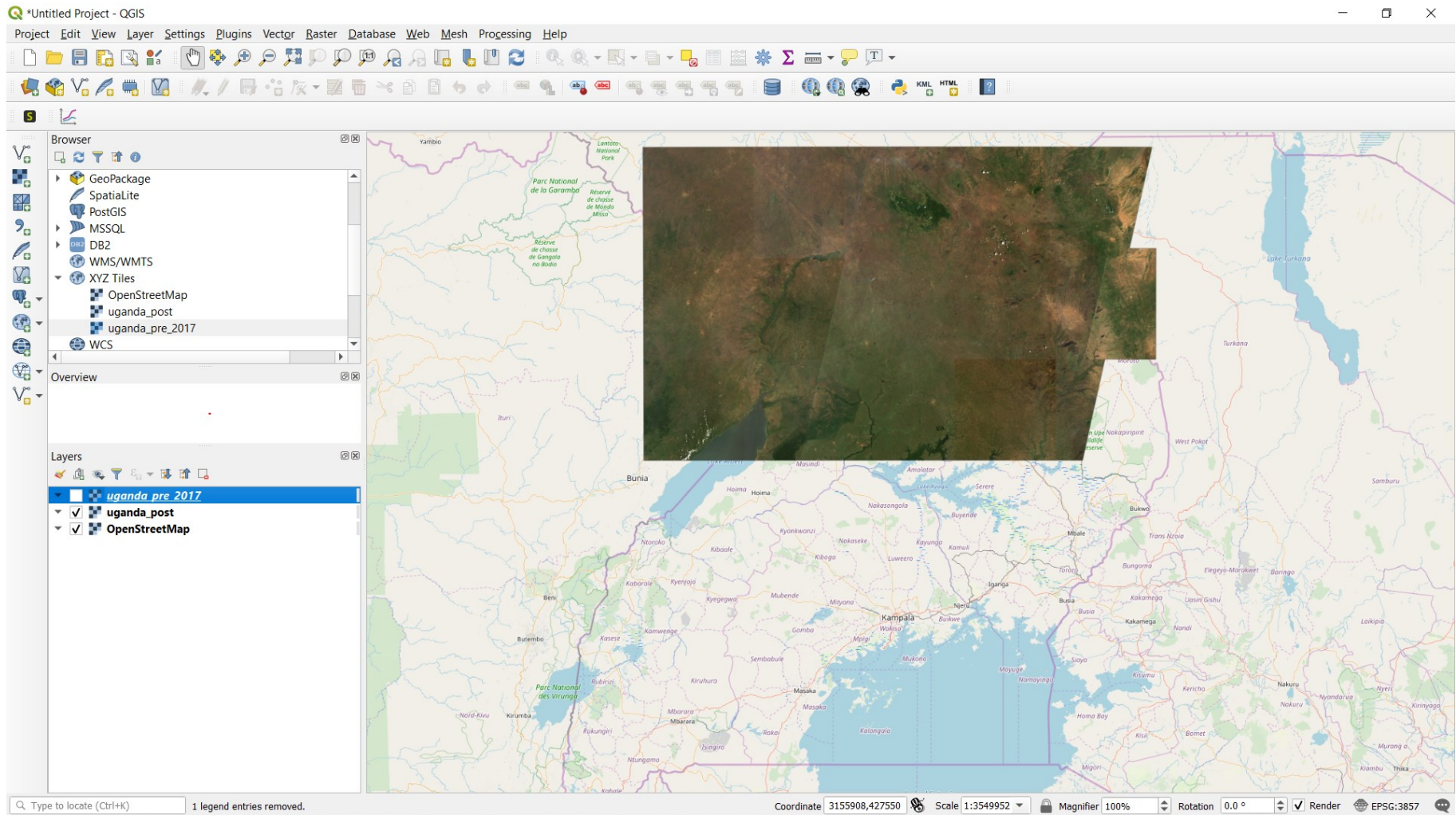
Figure 5.17 TMS for 2020 imported into QGIS.

The whole pipeline was incorporated into a Python script. At the moment, two versions are available, one for L2A products and one for L1C products. The two scripts are available in the Annexes.

The following scheme represents the whole pipeline and the software and libraries used.
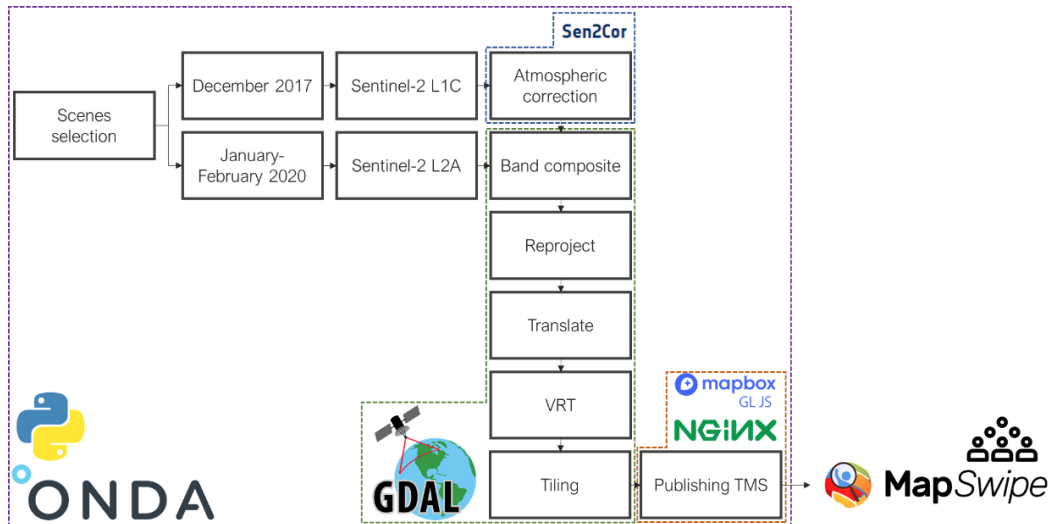
Figure 5.18 Developed pipeline.

Bugs

During the development of the full pipeline, some challenges were encountered in the ONDA infrastructure for which workarounds had to be developed. A brief list of these challenges and limitations are shown below:

- *Sen2Cor* processor has two separate versions (v2.5.5 and 2.8), for products generated with the 14.5 and 14.2 and for older than the 14.2 Products Specification Document used as input. In this specific case, since products from 2017 onwards were used as inputs, the newest version of *Sen2Cor* processor was sufficient for the development of the pipeline. An alternative script using the older version of *Sen2Cor* was also implemented for older products.

- During the development of the pipeline, a difference in the data structure of the products in SAFE format available in the ONDA infrastructure and the same ones available from other infrastructures (such as the ESA Open Access Hub platform) was noticed. This different consists in the two folders AUX_DATA and QI_DATA. While these are present as folders in the conventional products in SAFE format, the same are present as files in the products available in the ONDA catalogue. This difference prevents the normal functioning of the *Sen2Cor* processor. Hence, a workaround was

131

developed in the scripts in order to convert the files in folders. After this workaround, the *Sen2Cor* processor works properly.

- Another limitation was encountered when comparing the results of filtered products research in the online ONDA catalogue and in other catalogues (e.g. the ESA Open Access Hub). The research available at the ONDA online catalogue has some limitations, in that, once some filters are applied, it does not return the full list of products actually responding to that filter. This requires a careful approach when investigating the existence of optimal imagery over a selected AOI. This will be communicated to the ONDA development support.

Future developments

Some considerations on potentially interesting future developments of the current developed pipeline. The current pipeline requires a preliminary manual research of the products that want to be used for the mosaic and tiling. In addition, the mosaic developed is a simple composite mosaic obtained by juxtaposing the products. Future developments may include:

- An automatic procedure to search products in the ONDA DIAS catalogue based on filters such as dates of interest, area of interest and cloud cover.

- The development of algorithms similar to the ones already available in platforms such as Google Earth Engine to generate per-pixel composites applying some reducers to the imagery available. These algorithms allow to take into consideration per each tile the full availability of products given a time range, and generate a mosaic given by statistics calculated over the overlapping pixels (mean, median, highest, lowest, etc.). This allows getting rid of visual effects such as discontinuity across tiles and also to perform a more efficient cloud removal by investigating the available pixel free from clouds over the same imagery.

5.3.   Discussion and conclusions

This chapter investigated and illustrated a potential case study application for the newly available cloud infrastructures DIAS. Simultaneously, high-resolution open imagery such as Sentinel-2 were again the object of the application, lending themselves very well to the generation of large scale TMS to be provided to volunteer mappers for identification of displacement features by means of visual interpretation.

The MapSwipe campaign will be launched soon recurring to the TMS generated within a research group at the Oregon State University. Then it will be possible to quantitively investigate and measure the extent to which high-resolution

imagery were suitable for the visual detection of such features by a community of volunteer mappers.

In addition, it would be possible to explore what are the best options for the users to better identify a specific feature such as informal settlements from Sentinel-2 satellite imagery. To this end, indeed, a variety of RGB synthesis of different bands combinations and imagery enhancement techniques can be evaluated and tested. Some examples could be to provide false-colour composites, greenest composites (see Chapter 2), multi-temporal composites (see Chapter 3) and so on.

Chapter 6

6. Conclusions and future developments

Displacement is a globally recognized and growing event, whose magnitude is only expected to rise in the next years, following the increasing magnitude and impact of climate change, conflicts, disasters, and social and political dynamics, all of them interconnected in complex mutual feedbacks mechanisms. The strong need for information in displacement contexts is well supported by geospatial data, tools, and methods, as presented in Chapter 1. Despite the acknowledged power of geospatial data in displacement contexts and the widespread and growing availability of new data, tools, and methods, applications of high-resolution open satellite imagery in the context of displacement are relatively little explored in the scientific literature, as highlighted in Chapter 2. The reasons behind this are clear, imagery with resolution between four and thirty meters do not offer the possibility to map single infrastructures constituting displacement settings, which is one of the most common requests by humanitarian organization on the field. Anyhow, high-resolution open imagery, for instance from the Sentinel-2 mission, offers a wide range of benefits in comparison to the more commonly used very high-resolution imagery. Amongst them, higher temporal and spatial resolution and the lack of generalization issues. The potential uses and applications and the output that can be expected of high-resolution imagery for displacement contexts have been the key focus of this research.

In order to evaluate the potential uses and applications of high-resolution imagery for displacement contexts and the outputs that can potentially be expected from their use, the research schema already presented in Chapter 1 has been used to guide the explorative research. In the next sections, each branch of the research schema will be retraced and commented in the light of the obtained the results and conclusions achieved by this research.

The first branch of the research schema explored the quantitative use of high-resolution imagery, hence investigating image classification. It was decided to investigate both a direct application approach and an indirect one. The first was intended at verifying the usability of high-resolution imagery in mapping displacement settings themselves. The latter was intended at exploring indirect applications of high-resolution imagery, such as mapping indicators of displacement.
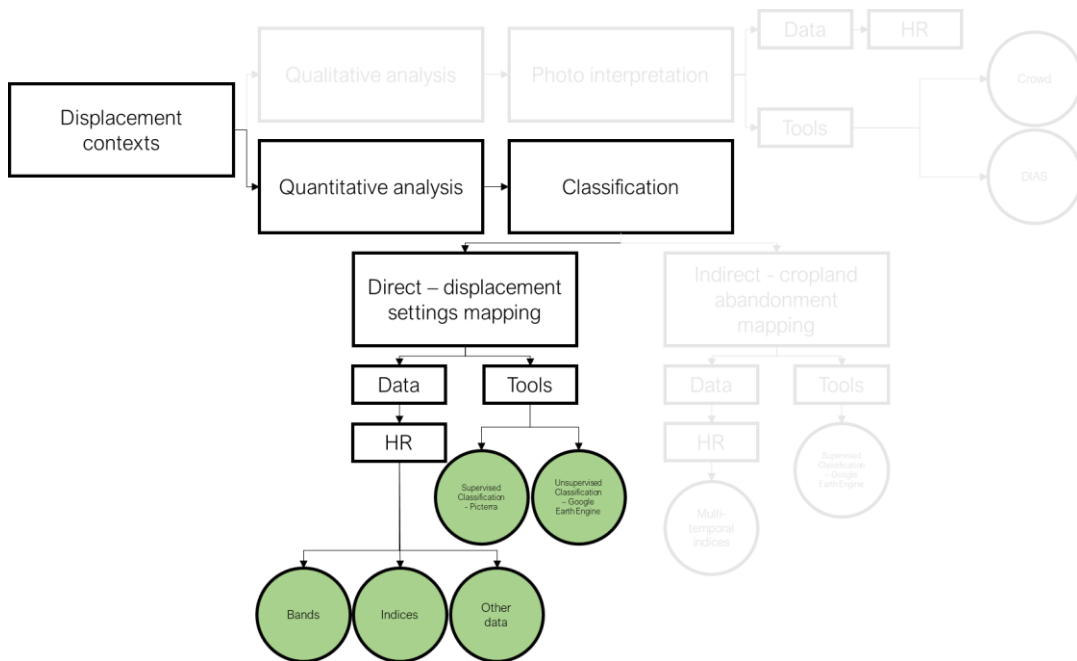
Figure 6.1 Research schema – direct displacement settings mapping.

When applied for directly classifying displacement settings, Sentinel-2 imagery has proved to be a potential useful source of information first in visually providing evidence of and then in locating and mapping the extent or outline of displacement settings, as illustrated in Chapter 3. As an exploratory exercise, the Picterra user-friendly platform providing a pre-trained deep learning algorithm has been preliminary tested. The Picterra platform serves as a handy tool in mapping many features that are similar from the shape and spectral characteristics point of view over large areas. The platform showed good results in identifying IDP camps in Iraq, where they are numerous over large areas and spectrally separable from the surrounding. This approach did not offer the basis for a more rigorous and systematic approach. A more conventional machine learning unsupervised algorithm has been implement rigorously and systematically for Sentinel-2 classifications in Google Earth Engine over four case studies. This approach provided good results in mapping the extent and outline of individual displacement settings. This is especially valid when spectral indices, such as NDVI and NDBI, and secondary dataset, such as night-time data, are implemented as classification properties in addition to the spectral bands of Sentinel-2 data. The four cases studies in Bangladesh, Uganda, Ethiopia, and Iraq revealed that unsupervised classification approaches provide acceptable overall validation accuracy metrics (up to 98%) in mapping the extent of displacement settings in four different geographical areas,

type of setting and surrounding environmental conditions. A generated displacement settings dataset served as validation dataset.
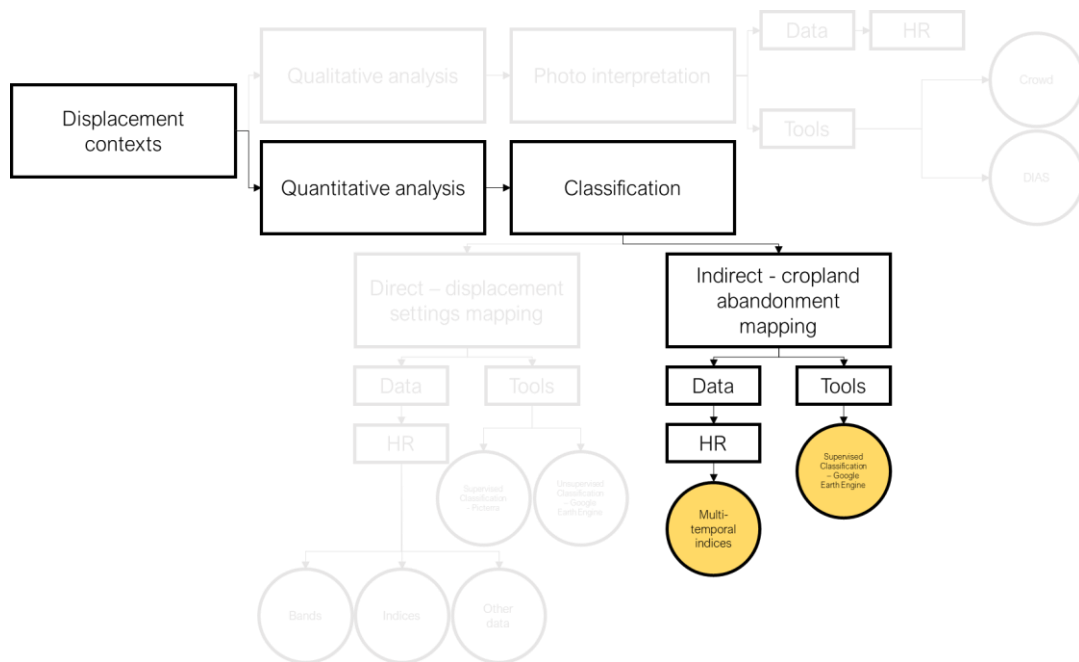
Figure 6.2 Research schema – indirect cropland abandonment mapping.

When applied for indirectly identifying displacement by mapping its potential indicators, high-resolution imagery proved to be qualitatively beneficial, even though achieved validation accuracies are low. Displacement is induced by and can induce itself a series of events. Amongst them, cropland abandonment was identified as a strong indicator of displacement in conflict affected areas in Chapter 4. Landsat-8 and Sentinel-2 time series of data have been implemented to map abandoned cropland in north-eastern Nigeria with a multi-temporal and stratified supervised classification approach in Google Earth Engine. This time, multi temporal NDVI stacks derived from Landsat-8 and Sentinel-2 imagery were used as input of a supervised machine learning algorithm. Overall validation accuracies resulting from the classification are low, nevertheless visual inspection of results suggested that cropland abandonment dynamics were captured, despite some relevant commission error. The overall approach could definitely be improved by improving both the training and the validation datasets.

Instead of exploiting imagery from a quantitative point of view, the second branch of the research schema explored a qualitative approach to high-resolution imagery. After assessing in Chapter 3 that high-resolution imagery can provide evidence in visually locating at least some of the existing displacement settings, photo interpretation methods can be implemented to take advantage of this type of data. The photo interpretation approach offered a case study application for one of the most recent cloud infrastructures, the ONDA Data Information and Access Service.
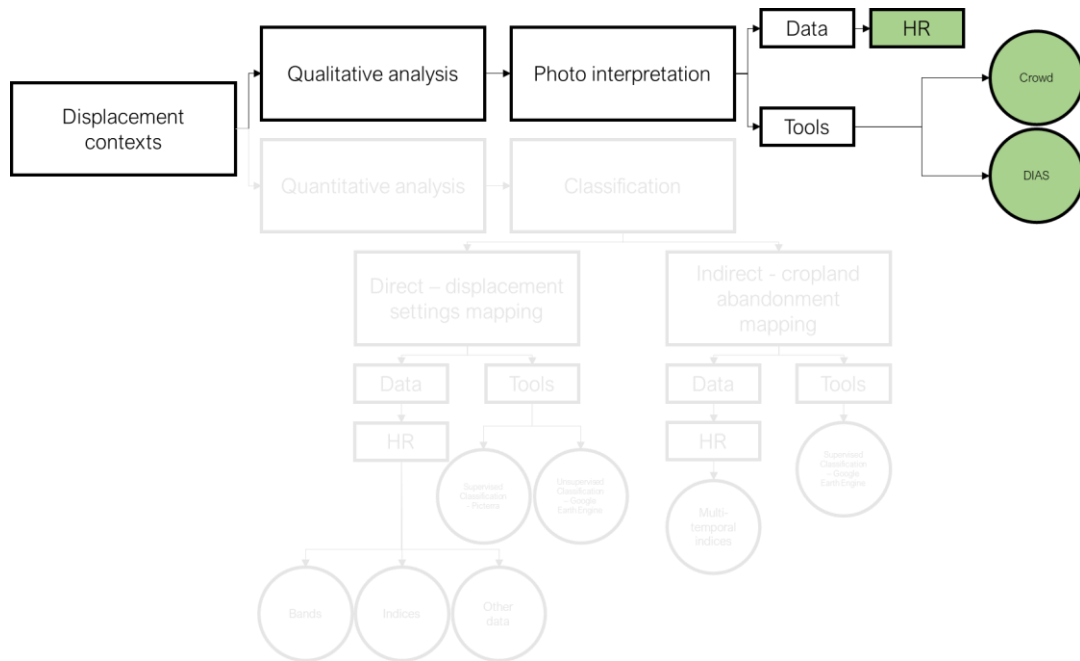
Figure 6.3 Research schema –photo interpretation.

In Chapter 5, hence, a crowdsourcing project aimed at crowd-mapping displacement settings across the world lend itself as a potential case study application of the ONDA Data Information and Access Service, one of the most recent cloud infrastructures leveraging the power of big data and cloud computing. A workable and reusable pipeline to retrieve large scale mosaics of Sentinel-2 imagery to be implemented as TMS in any application was developed entirely based on the ONDA DIAS. Once again, high-resolution open Sentinel-2 imagery provided a useful support and application, in this case when they are needed for visually interpreting imagery for displacement detection. The improvement and the investigation of the best imagery synthesis and composites for optimally enabling volunteer mappers to identify displacement features is one of the potential future developments of this research.

The transposal of the semi-automatic classification algorithms and monitoring tools investigated in Chapter 3 and Chapter 4 into the ONDA DIAS infrastructure is another potential future development of this research. By leveraging the described generated displacement dataset, it would be possible to develop a tool and a user interface for monitoring the evolution of displacement settings, providing the user with basic tools to perform simple image analysis, image composites or indices production. The end user would be provided with an assisted extraction of on-the-fly information. Being new imagery continuously uploaded in the ONDA DIAS catalogue, the developed tool could ingest the new imagery, identify, and monitor the features of interest with a high temporal resolution. Newly available software such as the Open Data Cube (https://www.opendatacube.org/) could also be explored in their potential to manage the large availability of data in a cloud infrastructure, easing the filtering of the catalogue itself.

137

7. References

Agenda for Humanity (2016). Initiative – Grand Bargain. Retrieved from https://www.agendaforhumanity.org/initiatives/3861

Bramhe, V. S., Ghosh, S. K. and Garg, P. K. (2018). Extraction of built-up areas using convolutional neural networks and transfer learning from sentinel-2 satellite images. *Int. Arch. Photogramm. Remote Sens. Spatial Inf.* Sci., XLII-3, 79-85, https://doi.org/10.5194/isprs-archives-XLII-3-79-2018

Çetinkaya C., Özceylan E., Erbaş M. and Kabak M. (2016). GIS-based fuzzy MCDA approach for siting refugee camp: A case study for southeastern Turkey. http://dx.doi.org/10.1016/j.ijdrr.2016.07.004

CM Toolkit (2015). CM TOOLKIT - Resources for practitioners working with displaced communities. Retrieved from http://cmtoolkit.org/

Copernicus (2013). Emergency Management Service – Mapping. EMSR024: Flooding of Refugee Camp in Mafraq. Retrieved from https://emergency.copernicus.eu/mapping/list-of-components/EMSR024

Copernicus (2017). The upcoming Copernicus Data and Information Access Services (DIAS). Retrieved from https://www.copernicus.eu/en/upcoming-copernicus-data-and-information-access-services-dias

Copernicus (2018). EMSR318: Monitoring Internally Displaced People sites in Afghanistan. Retrieved from https://emergency.copernicus.eu/mapping/list-of-components/EMSR318

Copernicus (2019). DIAS. Retrieved from https://www.copernicus.eu/en/access-data/dias

Copernicus (2019a). Emergency Management Service – Mapping. List of EMS Rapid Mapping Activations. Retrieved from https://emergency.copernicus.eu/mapping/list-of-activations-rapid

Copernicus (2020). EMSN-063. Crop change detection conflict areas, Nigeria - Final Report. Available at https://emergency.copernicus.eu/mapping/list-of-components/EMSN063

Copernicus (2020a). DIAS. Retrieved from https://www.copernicus.eu/en/access-data/dias

CSC (2020). Services for Research – Virtual datasets. Retrieved from https://research.csc.fi/virtual_rasters

Elia A., Balbo S. and Boccardo P. (2018). A quality comparison between professional and crowdsourced data in emergency mapping for potential cooperation of the services. *European Journal of Remote Sensing*, Volume 51, 2018 - Issue 1

ESA (2015). DOCUMENT - Copernicus Space Component Data Access Portfolio: Data Warehouse 2014 – 2020. Retrieved from https://spacedata.copernicus.eu/documents/12833/14545/DAP_Release_Phase_2_1.0_final

ESA (2020). Sentinel-2. Retrieved from https://earth.esa.int/web/sentinel/missions/sentinel-2

FAO/UNHCR (2016). Assessing woodfuel supply and demand in displacement settings, by D'Annunzio, R., Gianvenuti, A., Henry, M., Thulstrup, A. Rome, Italy.e.g.

FAO (2021). Crop Calendar - Nigeria. Retrieved from https://cropcalendar.apps.fao.org/#/home?id=NG&crops=

FEWS NET (2018). Nigeria Food Security Outlook. Retrieved from https://reliefweb.int/sites/reliefweb.int/files/resources/Main%20season%20harvests%20in%20much%20of%20the%20northeast%20have%20been%20severely%20limited%20by%20the%20ongoing%20conflict.pdf

FloodList (2016). Using Drones to Map and Model Flood Risks in Dar Es Salaam, Tanzania. Retrieved from http://floodlist.com/africa/using-drones-map-model-flood-risks-dar-es-salaam-tanzania

FloodList (2016a). Using Drones for Flood Disaster Risk Reduction, Assessment and Recovery. Retrieved from http://floodlist.com/protection/using-drones-for-disaster-risk-reduction-assessment-recovery

Forbes (2020). Satellite Mapping Aids North Korean Human Rights, But Caution Is Needed. Retrieved from https://www.forbes.com/sites/christinero/2020/02/01/satellite-mapping-aids-north-korean-human-rights-but-caution-is-needed/#33c1c6135523

Füreder P. (2015). Earth observation and GIS to support humanitarian operations in refugee/IDP camps. *Short Paper – Geospatial Data and Geographical*

Information Science Proceedings of the ISCRAM 2015 Conference - Kristiansand, May 24-27 Palen, Büscher, Comes & Hughes, eds.

GAEL (2017). Gael System. Retrieved from https://www.gael-systems.com/

GDAL (2019). GDAL – Documentation. Retrieved from https://gdal.org/

Goldblatt R., You W., Hanson G. and Khandelwal A.K. (2016). Detecting the Boundaries of Urban Areas in India: A Dataset for Pixel-Based Image Classification in Google Earth Engine. *Remote Sensing*. 2016, 8, 634; doi:10.3390/rs8080634

Goldblatt R., Stuhlmacherb M., Tellmanb B., Clintonc N., Hansona G., Georgescub M., Wangb C., Serrano-Candelad F., Khandelwale A. K., Chengb W. and Balling Jrb R. C. (2018). Using Landsat and nighttime lights for supervised pixel-based image classification of urban land cover. *Remote Sensing of Environment.* 205 (2018) 253–275

Goldblatt R., Jones N. and Mannix J. (2020). Assessing OpenStreetMap Completeness for Management of Natural Disaster by Means of Remote Sensing: A Case Study of Three Small Island States (Haiti, Dominica and St. Lucia). *Remote Sensing*. 2020, 12, 118; doi:10.3390/rs12010118

Google (2020). Machine Learning Crash Course. Retrieved from https://developers.google.com/machine-learning/crash-course

Google Earth Engine (2019). Google Earth Engine. Retrieved from https://earthengine.google.com/

Google Earth Engine (2019a). Earth Engine Data Catalog. Retrieved from https://developers.google.com/earth-engine/datasets/

Google Earth Engine (2021). Unsupervised Classification (clustering). Retrieved from https://developers.google.com/earth-engine/guides/clustering

Harvard Humanitarian Initiative (2011). Disaster Relief 2.0: The Future of Information Sharing in Humanitarian Emergencies. Washington, D.C. & Berkshire, UK: UN Foundation & Vodafone Foundation Technology Partnership

HDX (2019). The Humanitarian Data Exchange. Retrieved from https://data.humdata.org/

Herfort B., Li H., Fendrich S., Lautenbach S. and Zipf A. (2019). Mapping Human Settlements with Higher Accuracy and Less Volunteer Efforts by

Combining Crowdsourcing and Deep Learning. *Remote Sens*. 2019, 11, 1799; doi:10.3390/rs11151799

HOT (2017). Mapping the Missing Millions. Retrieved from https://www.hotosm.org/projects/mapping-the-missing-millions

Humanitarian OpenStreetMap Team (2019). Humanitarian OpenStreetMap Team. Retrieved from https://www.hotosm.org/

IMPACT (2021). Impact Initiatives. Retrieved from https://www.impact-initiatives.org/

Kemper T., Jenerowicz M., Gueguen L., Poli D. and Soille P. (2011). Monitoring changes in the Menik Farm IDP camps in Sri Lanka using multi-temporal very high-resolution satellite data. *International Journal of Digital Earth*, 4:sup1, 91-106, DOI: 10.1080/17538947.2010.512430

Konecny M., Zlataniva S. and Bandrova T.L. (2010). Geographic Information and Cartography for Risk and Crisis Management – Towards Better Solutions. *Lecture Notes in Geoinformation and Cartography*

Kumar L. and Mutanga O. (2018). Google Earth Engine Applications Since Inception: Usage, Trends, and Potential. *Remote Sens*. 2018, 10, 1509; doi:10.3390/rs10101509

L'Espresso (2009). Terremoto, il dolore e la rabbua. Retrieved from http://temi.repubblica.it/espresso-terremoto-in-abruzzo/tag/tendopoli/?refresh_ce

Lang, S., Tiede, D., Hölbling, D., Füreder, P. and Zeil, P. (2010). Earth observation (EO)-based ex post assessment of internally displaced person (IDP) camp evolution and population dynamics in Zam Zam, Darfur. *International Journal of Remote Sensing*. 31(21), 5709-5731

LearnOSM (2015). HOT Remote Response Guide. Retrieved from http://learnosm.org/en/coordination/HOT-Remote-Response-Guide/

Liedman (2020). The Hitchhacker's Guide To Tiled Maps. Retrieved from http://www.liedman.net/tiled-maps/

Logar T., Bullock J., Nemni E., Bromley L., Quinn J.A., Luengo-Oroz M. (2020). PulseSatellite: A tool using human-AI feedback loops for satellite image analysis in humanitarian contexts. Retrieved from https://www.researchgate.net/publication/338911694_PulseSatellite_A_tool_using_human-AI_feedback_loops_for_satellite_image_analysis_in_humanitarian_contexts

Lubej M., Aleksandrov M., Batic M., Kadunc M., Milkinski G., Peressutti D. and Zupanc A. (2018). Spatio-Temporal Deep Learning: Application to Land Cover Classification. *Living Planet Symposium*, DOI:10.13140/RG.2.2.17961.65128

Lubej M., Aleksandrov M., Batic M., Kadunc M., Milkinski G., Peressutti D. and Zupanc A. (2019). Spatio-Temporal Deep Learning. An Application to Land Cover Classification. Retrieved from https://sentinel-hub.com/sites/default/lps_2019_eolearn_TFCN.pdf

Mastrojeni G., Pasini A (2017). Effetto serra effetto guerra. Clima, conflitti, migrazioni: l'Italia in prima linea. Chiarelettere editore

Mapio (2021). No title. Retrieved from https://mapio.net/pic/p-60511792/

MapSwipe Back-End (2019). Installation. Retrieved from https://mapswipe-workers.readthedocs.io/en/dev/installation.html

Minghini M., Brovelli M.A. and Frassinelli F. (2018). An open source approach for the intrinsic assessment of the temporal accuracy, up-to-datedness and lineage of Openstreetmap. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume XLII-4/W8, 2018 FOSS4G 2018 – Academic Track, 29–31 August 2018, Dar es Salaam, Tanzania

Mooney D., French B. and Tomaszewski B. (2017). Development of a Block Level Geovisualization Tool of Zaatari Refugee Camp Community Asset Data. *IEEE Global Humanitarian Technology Conference (GHTC)*, October 2017, pp.1-8

MVAM (2018). Mapping our way towards Zero Hunger. Retrieved from http://mvam.org/2018/07/20/mapping-our-way-towards-zero-hunger-2/

Oloruntoba R., Banomyong R. (2018). Humanitarian logistics research for the care of refugees and internally displaced persons: A new area of research and a research agenda. *Journal of Humanitarian Logistics and Supply Chain Management*, Vol. 8 Issue: 3, pp.282-294, https:// doi.org/10.1108/JHLSCM-02-2018-0015

ONDA (2018). ONDA User Guides. Retrieved from https://www.onda-dias.eu/cms/knowledge-base/

Open Data for Resilience Initiative (2018). Uganda Bureau of Statistics engages in open mapping for resilience with the OpenStreetMap community. Retrieved from https://opendri.org/uganda-open-mapping-for-resilience/

OpenStreetMap Wiki (2019). MapSwipe. Retrieved from https://wiki.openstreetmap.org/wiki/MapSwipe

Patel N., Angiuli E., Gamba P., Gaughan A., Lisin G., Stevens F., Tatem A. and Trianni G. (2014). Multitemporal settlement and population mapping from Landsat using Google Earth Engine. *International Journal of Applied Earth Observation and Geoinformation* 35 (2015) 199–208

Picterra (2020). Picterra - Humanitarian and disaster risk management. How to react timely and effectively after a natural disaster? Retrieved from https://picterra.ch/applications/humanitarian-and-disaster-risk-management/

Quinn J. A., Nyhan M. M., Navarro C., Coluccia D., Bromley L. and Luengo-Oroz M. (2018). Humanitarian applications of machine learning with remote-sensing data: review and case study in refugee settlement mapping. *Phil. Trans.* R. Soc. A 376: 20170363. http://dx.doi.org/10.1098/rsta.2017.0363

REACH (2013). Informal Tented Settlements in Jordan: A Multi-Sector, Baseline Assessment. Retrieved from http://www.reachresourcecentre.info/system/files/resource-documents/marie.loeb-28012014-045043-reach_unicef_informal_tented_settlements_in_jordan_-_a_multi-sector_baseline_assessment_dec_2013.pdf

REACH (2015). REACH Rapid Assessment identifies Priority Winter Needs of Syrians living in Za'atari Camp. Retrieved from http://www.reach-initiative.org/reach-rapid-assessment-identifies-priority-winter-needs-of-syrians-living-in-zaatari-camp

REACH (2018). Joint Multi-Sector Needs Assessment - Identifying humanitarian needs among refugee and host community populations in Uganda. Retrieved from http://www.reachresourcecentre.info/system/files/resource-documents/reach_uga_msna_report_aug2018.pdf

REACH (2018a). Bangladesh: REACH supports monsoon season emergency preparedness of Rohingya camps. Retrieved from http://www.reach-initiative.org/bangladesh-reach-supports-monsoon-season-emergency-preparedness-of-rohingya-camps

Reliefweb (2015). Khanke Camp profile. Kabarto 1 Camp Profile. Kabarto 2 Camp Profile. Shariya Camp Profile. Retrieved from https://reliefweb.int/report/iraq/khanke-camp-profile-dahuk-governorate-iraq-september-2015-enar https://reliefweb.int/report/iraq/khanke-camp-profile-dahuk-governorate-iraq-september-2015-enar https://reliefweb.int/report/iraq/kabarto-2-camp-profile-dahuk-governorate-iraq-september-2015

https://reliefweb.int/report/iraq/shariya-camp-profile-dahuk-governorate-iraq-september-2015

Reliefweb (2017). Bangladesh – Rohingya Crisis: Mapping infrastructure and services in refugee camps and sites. Retrieved from https://reliefweb.int/report/bangladesh/bangladesh-rohingya-crisis-mapping-infrastructure-and-services-refugee-camps-and

Reliefweb (2018). Record Rainfall Devastates IDP Camps in Northern Syria. Retrieved from https://reliefweb.int/report/syrian-arab-republic/record-rainfall-devastates-idp-camps-northern-syria

Reliefweb (2020). Ethiopia Operation Kobe - Camp Profile, November 2020. Retrieved from https://reliefweb.int/report/ethiopia/ethiopia-operation-kobe-camp-profile-november-2020

Reliefweb (2020a). Iraq: IDP Camp Directory - Comparative Dashboard & Camp Profiles Round XIII, February-March 2020. Retrieved from https://reliefweb.int/report/iraq/iraq-idp-camp-directory-comparative-dashboard-camp-profiles-round-xiii-february-march

Reuters (2007). Amnesty keeps an eye on Darfur with satellite images. *Technology news*. Retrieved from https://www.reuters.com/article/us-sudan-darfur-amnesty/amnesty-keeps-an-eye-on-darfur-with-satellite-images-idUSN0530427120070606

Reuters Graphics (2017). The Rohingya Crisis - Life in the camps. Retrieved from http://fingfx.thomsonreuters.com/gfx/rngs/MYANMAR-ROHINGYA/010051VB46G/index.html

Rodríguez-Espíndola O., Albores P. and Brewster C. (2016). GIS and Optimisation: Potential Benefits for Emergency Facility Location in Humanitarian Logistics. *Geosciences 2016*, 6, 18; doi:10.3390/geosciences6020018

Rumora L., Miler M. and Medak D. (2020). Impact of Various Atmospheric Corrections on Sentinel-2 Land Cover Classification Accuracy Using Machine Learning Classifiers. *ISPRS Int. J. Geo-Inf.* 2020, 9, 277; doi:10.3390/ijgi9040277

Sammonds P. R., Ahmed B., Rahman M. S. and Islam, R. (2018). Developing a Landslide Early Warning System for the Rohingya Refugees in Cox's Bazar, Bangladesh. Abstract retrieved from http://adsabs.harvard.edu/abs/2018AGUFMGC23C..11S

Schön A., Al-Saadi S., Grubmueller J. and Schumann-Bölsche D. (2018). Developing a camp performance indicator system and its application to Zaatari,

Jordan. *Journal of Humanitarian Logistics and Supply Chain Management* Vol. 8 No. 3, 2018 pp. 346-373 Emerald Publishing Limited 2042-6747 DOI 10.1108/JHLSCM-10-2017-0047

Scopus (2021). Advanced search. Retrieved from https://www.scopus.com/search/form.uri?display=advanced&zone=header&origin =

Seed (2015). Cutting down trees for survival: How women can provide an alternative source of energy. Retrieved from https://www.seed.uno/articles/blog/cutting-down-trees-for-survival-how-women-can-provide-an-alternative-source-of-energy

Seifert L., Kunz N. and Gold S., (2018). Humanitarian supply chain management responding to refugees: a literature review. *Journal of Humanitarian Logistics and Supply Chain Management*, Vol. 8 Issue: 3, pp.398-426, https://doi.org/10.1108/JHLSCM-07-2017-0029

Soden R., Palen L. (2014). From crowdsourced mapping to community mapping: The post-earthquake work of OpenStreetMap Haiti. COOP 2014-Proceedings of the 11th International Conference on the Design of Cooperative Systems, 27-30 May 2014, Nice (France).

SpaceNet (2020). SpaceNet on AWS. Retrieved from https://spacenetchallenge.github.io/datasets/datasetHomePage.html

Sphere (2018). The Sphere Handbook – Humanitarian Charter and Minimum Standards in Humanitarian Response. Retrieved from https://spherestandards.org/wp-content/uploads/Sphere-Handbook-2018-EN.pdf

Stefanakis E. (2017). Web mercator and raster tile maps: two cornerstones of online map service providers. Geomatica. Retrieved from https://www.researchgate.net/publication/321064657_Web_mercator_and_raster_tile_maps_two_cornerstones_of_online_map_service_providers

The Guardian (2015). Our life in the Zaatari refugee camp: no electricity, no space to sleep, no escape. Retrieved from https://www.theguardian.com/commentisfree/2015/sep/14/life-refugee-camp-syrian-family-jordan-escape

The Guardian (2018). Inside the world's largest refugee camp: 'We just want to go home'. Retrieved from https://www.theguardian.com/world/2018/may/19/inside-world-largest-refugee-camp-uganda-south-sudan-civil-war

The Guardian (2018a). Start of rainy season exposes risks for Rohingya refugees in Bangladesh. Retrieved from https://www.theguardian.com/world/2018/apr/18/first-monsoon-rain-exposes-risks-for-rohingya-refugees-in-bangladesh

TheInterpreter (2018). The Rohingya are stuck. Retrieved from https://www.lowyinstitute.org/the-interpreter/rohingya-are-stuck

Themenbuero (2021). KRI: Establishing good PR practice. Themenbuero – Visual Storytelling. Retrieved from https://www.themenbuero.de/kri-establishing-good-pr-practice/

The New York Times (2018). In Uganda's Refugee Camps, South Sudanese Children Seek the Families They've Lost. Retrieved from https://www.nytimes.com/2018/08/09/world/africa/south-sudan-refugee-children-uganda.html

Thinking Machines (2020). Mapping New Informal Settlements for Humanitarian Aid through Machine Learning. Retrieved from https://stories.thinkingmachin.es/mapping-new-informal-settlements/?utm_source=socmed&utm_medium=fb&utm_campaign=immap_firs tpost&fbclid=IwAR3jxFl6XjZ_CjVxsft9d8DFEDfUAlsVzRnD4tZFCgJoof6h3w 13OXDBuWQ

Thomas A.S. and Kopczak L.R. (2005). From Logistics to Supply Chain Management: The Path Forward in the Humanitarian Sector. Fritz Institute, San Francisco, CA

UNHCR (2015). The Emergency Handbook - Site planning for camps. Retrieved from https://emergency.unhcr.org/entry/254415/site-planning-for-camps

UNHCR (2015a). The Emergency Handbook - Managing and supporting spontaneous settlements. Retrieved from https://emergency.unhcr.org/entry/217722/managing-and-supporting-spontaneous-settlements

UNHCR (2015b). The Emergency Handbook - Camp coordination, camp management (CCCM). Retrieved from https://emergency.unhcr.org/entry/254805/camp-coordination-camp-management-cccm

UNHCR (2015c). The Emergency Handbook - Settlement in urban areas. Retrieved from https://emergency.unhcr.org/entry/253592/settlement-in-urban-areas

UNHCR (2015d). Emergency Handbook - UNHCR guide to agile, effective and community-based humanitarian emergency responses. Retrieved from https://emergency.unhcr.org/

UNHCR (2018). Forced displacement at record 68.5 million. Retrieved from https://www.unhcr.org/news/stories/2018/6/5b222c494/forced-displacement-record-685-million.html

UNHCR (2018a). Ethiopia. Retrieved from https://www.unhcr.org/ethiopia.html

UNHCR (2019). Figures at a Glance. Retrieved from https://www.unhcr.org/figures-at-a-glance.html

UNHCR (2019a). Environment, Disasters and Climate Change. Retrieved from https://www.unhcr.org/environment-disasters-and-climate-change.html

UNHCR (2019b). Rohingya emergency. Retrieved from https://www.unhcr.org/rohingya-emergency.html

UNHCR (2021). Operational Portal – Refugee Situations – Ethiopia. Retrieved from https://data2.unhcr.org/en/country/eth/161

UNHCR (2021a). Uganda Comprehensive Refugee Response Portal. Retrieved from https://data2.unhcr.org/en/country/uga

UNHCR (2021b). Operational Portal – IDP Situations – Iraq CCCM Cluster. Retrieved from https://data2.unhcr.org/en/situations/iraq_cccm#:~:text=Iraq%20is%20experiencing%20ongoing%20displacement,areas%20of%20origin%20have%20occurred.

UNHCR (2021c). Iraq. Khanke IDP Camp shelters 10,000 displaced Iraqis. Retrieved from https://www.unhcr.org/cn/%E8%81%94%E5%90%88%E5%9B%BD%E9%9A%BE%E6%B0%91%E7%BD%B2%E7%AE%80%E4%BB%8B/iraq-khanke-idp-camp-shelters-10000-displaced-iraqis-2

UNITAR (2018). Satellite Detected Waters Evolution by Camp Extent in Cox's Bazar District, Chittagong Division, Bangladesh. Retrieved from https://unitar.org/unosat/node/44/2820?utm_source=unosat-unitar&utm_medium=rss&utm_campaign=maps

UN-SPIDER (2018). Copernicus launches DIAS services during 20th anniversary celebrations. Knowledge portal. Retrieved from http://www.un-

spider.org/news-and-events/news/copernicus-launches-dias-services-during-20th-anniversary-celebrations

USA for UNHCR (2019). What is a Refugee Camp? Retrieved from https://www.unrefugees.org/refugee-facts/camps/

USA for UNHCR (2019). What is a refugee camp? Retrieved from https://www.unrefugees.org/refugee-facts/camps/

USAID (2020). Iraq – Complex Emergency. Retrieved from https://reliefweb.int/sites/reliefweb.int/files/resources/2020_12_16%20USG%20Iraq%20Complex%20Emergency%20Fact%20Sheet%20%231.pdf

Voigt, S., Schoepfer, E., Fourie, C. and Mager, A. (2014). Towards semi-automated satellite mapping for humanitarian situational awareness. *Global Humanitarian Technology Conference (GHTC), 2014 IEEE*, 412-416

Voigt S. et al. (2016). Global trends in satellite-based emergency mapping. Science 353 (6296), 247-252. [doi: 10.1126/science.aad8728]

Wendt et al. (2015). Using Remote Sensing and GIS to Support Drinking Water Supply in Refugee/IDP Camps. *GI_Forum – Journal for Geographic Information Science*, 1-2015

Westrope C. and Poisson E. (2013). Using GIS as a planning and coordination tool in refugee camps in South Sudan. Humanitarian Practice Network. Retrieved from https://odihpn.org/magazine/using-gis-as-a-planning-and-coordination-tool-in-refugee-camps-in-south-sudan/

WFP (2021). Uganda. Retrieved from https://www.wfp.org/countries/uganda

WHO (2019). Displaced people. Retrieved from https://www.who.int/environmental_health_emergencies/displaced_people/en/

Wikipedia (2019). Tiled web map. Retrieved from https://en.wikipedia.org/wiki/Tiled_web_map

Wikipedia (2020a). Slippy map tilenames. Rettrieved from https://wiki.openstreetmap.org/wiki/Slippy_map_tilenames

Wikipedia (2020b). Web server. Rettrieved from https://en.wikipedia.org/wiki/Web_server

Witmer F. D. W. (2015). Remote sensing of violent conflict: eyes from above. *International Journal of Remote Sensing*. 36:9, 2326-2352, DOI:10.1080/01431161.2015.1035412

WFP (2020). Satellite imagery in conflict-affected areas - How technology can support WFP emergency response. Available at https://www.wfp.org/publications/wfp-mali-satellite-imagery

World Food Programme Insight (2018). Meet the new and extended ADAM - The powerful weapon that triggers emergency preparedness for a timely response. Retrieved from https://insight.wfp.org/meet-the-new-and-extended-adam-e15956b7669d

Xiong J., Thenkabail P., Tilton J., Gumma M., Teluguntla P., Oliphant A., Congalton R., Yadav K. and Gorelick N. (2017). Nominal 30-m Cropland Extent Map of Continental Africa by Integrating Pixel-Based and Object-Based Algorithms Using Sentinel-2 and Landsat-8 Data on Google Earth Engine. Remote Sens. 2016, 9, 1065; doi:10.3390/rs9101065. Available at https://www.mdpi.com/2072-4292/9/10/1065/htm

Yin H., et al., Brandão A., Buchner J., Helmers D., Iuliano B., Kimambo N., Lewińska K., Razenkwova E., Rizayeva A., Rogova N., Spawn S. and Xie Y.,2020. Monitoring cropland abandonment with Landsat time series. Remote Sensing of Environment 246 (2020) 111873. Available at https://www.sciencedirect.com/science/article/pii/S0034425720302431

ANNEXES

Error matrices for the displacement settings classifications

Bangladesh **2019**
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	471390	12381	483771	
Actual	Camp	52911	135296	188207	
	Overall	524301	147677	671978	
		Omission	Comission	PA	UA
	Non-camp	0.025592687	0.100917221	0.974407313	0.899082779
	Camp	0.281131945	0.083838377	0.718868055	0.916161623
	Overall	0.902836105			

Bangladesh **2019**
Yearly median of NDVI, NDBI

		Predicted			
		Non-camp	Camp		
	Non-camp	420140	63631	483771	
Actual	Camp	40129	148078	188207	
	Overall	460269	211709	671978	
		Omission	Comission	PA	UA
	Non-camp	0.131531241	0.087185972	0.868468759	0.912814028
	Camp	0.213217362	0.300558786	0.786782638	0.699441214
	Overall	0.845590183			

Bangladesh **2019**
Yearly median of NDVI, NDBI, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	475950	7821	483771	
Actual	Camp	72715	115492	188207	
	Overall	548665	123313	671978	
		Omission	Comission	PA	UA
	Non-camp	0.01616674	0.132530779	0.98383326	0.867469221
	Camp	0.386356512	0.06342397	0.613643488	0.93657603
	Overall	0.880150838			

Bangladesh **2019**
Yearly median of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12

		Predicted			
		Non-camp	Camp		
	Non-camp	464195	19576	483771	
Actual	Camp	89801	98406	188207	
	Overall	553996	117982	671978	
		Omission	Comission	PA	UA
	Non-camp	0.040465427	0.162096838	0.959534573	0.837903162
	Camp	0.477139533	0.165923615	0.522860467	0.834076385
	Overall	0.837231278			

Bangladesh **2019**
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	471819	11952	483771	
Actual	Camp	54805	133402	188207	
	Overall	526624	145354	671978	
		Omission	Comission	PA	UA
	Non-camp	0.024705904	0.104068557	0.975294096	0.895931443
	Camp	0.291195333	0.082226839	0.708804667	0.917773161
	Overall	0.900655974			

Bangladesh **2019**
Yearly greenest NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	471986	11785	483771	
Actual	Camp	89250	98957	188207	
	Overall	561236	110742	671978	
		Omission	Comission	PA	UA
	Non-camp	0.033077853	0.070694683	0.966922147	0.929305317
	Camp	0.458462674	0.070694683	0.541537326	0.929305317
	Overall	0.908108961			

VALIDATION ACCURACY

Uganda	2017			
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B4, B7, B11, B12				

		Predicted		
		Non-camp	Camp	
	Non-camp	3897560	188120	4085680
Actual	Camp	258869	179304	438173
	Overall	4156429	367424	4523853
		Omission	Comission	PA	UA
	Non-camp	0.046043743	0.062281588	0.953956257	0.937718412
	Camp	0.590791765	0.511997039	0.409208235	0.488002961
	Overall	0.901192855			

Uganda	2017			
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B4, B7, B11, B12, VIIRS				

		Predicted		
		Non-camp	Camp	
	Non-camp	3940795	144885	4085680
Actual	Camp	287104	151069	438173
	Overall	4227899	295954	4523853
		Omission	Comission	PA	UA
	Non-camp	0.035461661	0.067907015	0.964538339	0.932092985
	Camp	0.655229784	0.48955243	0.344770216	0.51044757
	Overall	0.904508612			

Uganda	2017			
Yearly greenest of NDVI, yearly median NDBI				

		Predicted		
		Non-camp	Camp	
	Non-camp	3911059	174621	4085680
Actual	Camp	321332	116841	438173
	Overall	4232391	291462	4523853
		Omission	Comission	PA	UA
	Non-camp	0.042739764	0.075922097	0.957260236	0.924077903
	Camp	0.733345049	0.599120983	0.266654951	0.400879017
	Overall	0.890369338			

Uganda	2017			
Yearly median of NDVI, NDBI, B4, B7, B11, B12, VIIRS				

		Predicted		
		Non-camp	Camp	
	Non-camp	3775354	310326	4085680
Actual	Camp	296992	141181	438173
	Overall	4072346	451507	4523853
		Omission	Comission	PA	UA
	Non-camp	0.075954553	0.07292897	0.924045447	0.92707103
	Camp	0.677796213	0.687311603	0.322203787	0.312688397
	Overall	0.865752048			

Uganda	2017			
Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12				

		Predicted		
		Non-camp	Camp	
	Non-camp	4003503	82177	4085680
Actual	Camp	309546	128627	438173
	Overall	4313049	210804	4523853
		Omission	Comission	PA	UA
	Non-camp	0.020113421	0.071769646	0.979886579	0.928230354
	Camp	0.706446997	0.389826569	0.293553003	0.610173431
	Overall	0.913409432			

Uganda	2017			
Yearly greenest of B4, B7, B11, B12				

		Predicted		
		Non-camp	Camp	
	Non-camp	4035638	50042	4085680
Actual	Camp	341707	96466	438173
	Overall	4377345	146508	4523853
		Omission	Comission	PA	UA
	Non-camp	0.012248145	0.078062616	0.987751855	0.921937384
	Camp	0.779844947	0.341564966	0.220155053	0.658435034
	Overall	0.913403685			

VALIDATION ACCURACY

	Ethiopia	2019		
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B8A, B11, B12, VIIRS				
		Predicted		
		Non-camp	Camp	
	Non-camp	358854	12603	371457
Actual	Camp	26760	32837	59597
	Overall	385614	45440	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.033928557	0.069395821	0.966071443	0.930604179
	Camp	0.44901589	0.277354754	0.55098411	0.722645246
	Overall	0.908681975			

	Ethiopia	2019		
Yearly greenest of NDVI, NDBI, B8A, B11, B12, VIIRS				
		Predicted		
		Non-camp	Camp	
	Non-camp	359170	12287	371457
Actual	Camp	27323	32274	59597
	Overall	386493	44561	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.033077853	0.070694683	0.966922147	0.929305317
	Camp	0.458462674	0.275734387	0.541537326	0.724265613
	Overall	0.908108961			

	Ethiopia	2019		
Yearly greenest of NDVI, yearly median NDBI				
		Predicted		
		Non-camp	Camp	
	Non-camp	286655	84802	371457
Actual	Camp	35235	24362	59597
	Overall	321890	109164	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.228295604	0.10946286	0.771704396	0.89053714
	Camp	0.591221035	0.77683119	0.408778965	0.22316881
	Overall	0.721526769			

	Ethiopia	2019		
Yearly greenest of NDVI, yearly median NDBI, VIIRS				
		Predicted		
		Non-camp	Camp	
	Non-camp	353652	17805	371457
Actual	Camp	24092	35505	59597
	Overall	377744	53310	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.04793287	0.063778644	0.95206713	0.936221356
	Camp	0.404248536	0.333989871	0.595751464	0.666010129
	Overall	0.902803361			

	Ethiopia	2019		
Yearly greenest of B2, B3, B4, B5, B6, B7, B8, B8A, B11, B12				
		Predicted		
		Non-camp	Camp	
	Non-camp	302220	69237	371457
Actual	Camp	35331	24266	59597
	Overall	337551	93503	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.186393041	0.104668628	0.813606959	0.895331372
	Camp	0.592831854	0.740478915	0.407168146	0.259521085
	Overall	0.757413224			

	Ethiopia	2019		
Yearly median of NDVI, NDBI, B8A, B11, B12, VIIRS				
		Predicted		
		Non-camp	Camp	
	Non-camp	357267	14190	371457
Actual	Camp	26145	33452	59597
	Overall	383412	47642	431054
		Omission	*Comission*	*PA*	*UA*
	Non-camp	0.038200922	0.068190354	0.961799078	0.931809646
	Camp	0.438696579	0.297846438	0.561303421	0.702153562
	Overall	0.906427037			

Iraq 2019
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	1160605	6125	1166730	
Actual	Camp	18100	30398	48498	
	Overall	1178705	36523	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.005249715	0.015355835	0.994750285	0.984644165
	Camp	0.373211266	0.167702544	0.626788734	0.832297456
	Overall	0.980065469			

Iraq 2019
Yearly median of NDVI, NDBI, B2, B3, B4, B5, B11, B12

		Predicted			
		Non-camp	Camp		
	Non-camp	1151300	15430	1166730	
Actual	Camp	16870	31628	48498	
	Overall	1168170	47058	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.013224996	0.014441391	0.986775004	0.985558609
	Camp	0.347849396	0.327893238	0.652150604	0.672106762
	Overall	0.973420626			

Iraq 2019
Yearly greenest of NDVI, yearly median NDBI, yearly greenest B2, B3, B4, B5, B11, B12, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	1142820	23910	1166730	
Actual	Camp	7771	40727	48498	
	Overall	1150591	64637	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.020493173	0.00675392	0.979506827	0.99324608
	Camp	0.160233412	0.36991197	0.839766588	0.63008803
	Overall	0.973929995			

Iraq 2019
Yearly median of NDVI, NDBI

		Predicted			
		Non-camp	Camp		
	Non-camp	952141	214589	1166730	
Actual	Camp	14568	33930	48498	
	Overall	966709	248519	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.183923444	0.015069685	0.816076556	0.984930315
	Camp	0.300383521	0.863471203	0.699616479	0.136528797
	Overall	0.811428802			

Iraq 2019
Yearly median of NDVI, NDBI, VIIRS

		Predicted			
		Non-camp	Camp		
	Non-camp	1088678	78052	1166730	
Actual	Camp	24795	23703	48498	
	Overall	1113473	101755	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.066898083	0.022268165	0.933101917	0.977731835
	Camp	0.511258196	0.76705813	0.488741804	0.23294187
	Overall	0.915368145			

Iraq 2019
Yearly median of B2, B3, B4, B5, B11, B12

		Predicted			
		Non-camp	Camp		
	Non-camp	1155537	11193	1166730	
Actual	Camp	21981	26517	48498	
	Overall	1177518	37710	1215228	
		Omission	Comission	PA	UA
	Non-camp	0.009593479	0.018667231	0.990406521	0.981332769
	Camp	0.453235185	0.29681782	0.546764815	0.70318218
	Overall	0.972701419			

GEE script for k-means classification on displacement settings (Sentinel-2)

```
//////////////////////////////////////////////////
//// SET IMPORTS AND VISUALIZATION PARAMETERS ////
//////////////////////////////////////////////////

var                              camps                              =
ee.FeatureCollection("users/agataelia1991/IRQ_20190831_name_dissolve_sample"),
imageVisParam                                                       =
{"opacity":1,"bands":["B4","B3","B2"],"min":497.48,"max":3448.52,"gamma":2.013},
imageVisParam2                                                      =
{"opacity":1,"bands":["B4_median","B3_median","B2_median"],"min":696.7225,"max":42
77.4025,"gamma":1.1},
imageVisParam3                                                      =
{"opacity":1,"bands":["VIIRS_median"],"min":0.14514687657356262,"max":1.1310894489
28833,"gamma":1.1},
imageVisParam4                                                      =
{"opacity":1,"bands":["TXT_median"],"min":5.348189830780029,"max":5.60964202880859
4,"gamma":0.746};


//////////////////////////////////////////////////
//// SET PARAMETERS, FUNCTIONS AND ADD AND INSPECT DATA ////
//////////////////////////////////////////////////

// Center the map over AOI
Map.centerObject(camps);
var aoi = camps.geometry().buffer(1000).bounds();

// Define roi
var camp = (ee.Feature(camps.geometry())).set('type', 'camp', 'class', 1);
var other = (ee.Feature(aoi.difference(camps, 10))).set('type', 'non-camp', 'class',
0);
var roi = ee.FeatureCollection([camp, other]);

// Define years
var year = 2019;
var year_string = '2019';
var years = [2016, 2017, 2018, 2019, 2020];

// Define bands
var bands = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
var bands_median = ['B2_median', 'B3_median', 'B4_median', 'B5_median', 'B6_median',
'B7_median', 'B8_median', 'B8A_median', 'B11_median', 'B12_median'];

// Define a list of Sentinel-2 wavelengths for charts X-axis labels
var wavelengths = [0.4966, 0.560, 0.6645, 0.7039, 0.7402, 0.7825, 0.8351, 0.8648,
1.6137, 2.2024]

// Define a function that rescales Sentinel-2 collection to reflectance values
var rescale = function(image){
  var rescaledImage = image.multiply(0.0001);
  return rescaledImage.copyProperties(image, image.propertyNames());
};

// Define a function that rescale images to 10m
var rescale10m = function(image){
  var image_10m = image.resample('bilinear').reproject({
  crs: 'EPSG:4326',
  scale: 10
  });
  return image_10m;
};

// Define a function that adds an NDVI band to a Sentinel-2 image
var addNDVI = function(image) {
  var ndvi = image.normalizedDifference(['B8', 'B4']).rename('NDVI');
```

```
    return image.addBands(ndvi);
};

// Define a function that adds an NDBI band at 10m to a Sentinel-2 image
var addNDBI = function(image){
  var band8 = image.select('B8');
  var band11 = image.select('B11');
  var band11_10m = band11.resample('bilinear').reproject({
  crs: band8.projection().crs(),
  scale: 10
  });
  var ndbi = ((band11_10m.subtract(band8))
  .divide(band11_10m.add(band8))).rename('NDBI');
  return image.addBands(ndbi);
};

//////////////////////////////////////////////////
//// CREATE COLLECTIONS ////
//////////////////////////////////////////////////

// Define and filter S2 collection based on AOI and map the addNDVI and addNDBI and
rescale
var collection = ee.ImageCollection('COPERNICUS/S2')
  .select(bands)
  .filterBounds(aoi)
  .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 5))
  .map(addNDVI)
  .map(addNDBI)
  .map(rescale10m);

// Define and filter VIIRS collection based on AOI rescale
var collection_viirs = ee.ImageCollection('NOAA/VIIRS/DNB/MONTHLY_V1/VCMCFG')
    .select('avg_rad')
    .filterBounds(aoi)
    .map(rescale10m);

// Create image collection of yearly greenests
var collection_yearly_greenest = ee.ImageCollection.fromImages(
  years.map(function(y) {
    var yearly_greenest = collection
        .filter(ee.Filter.calendarRange(y, y, 'year'))
        .qualityMosaic('NDVI')
        .clip(aoi)
        .set('name', 'greenest')
        .set('year', y);
    return yearly_greenest;
}));

print(collection_yearly_greenest);

// Create image collection of yearly medians
var collection_yearly_median = ee.ImageCollection.fromImages(
  years.map(function(y) {
    var yearly_median = collection
        .filter(ee.Filter.calendarRange(y, y, 'year'))
        .reduce(ee.Reducer.median())
        .clip(aoi)
        .set('name', 'median')
        .set('year', y);
    return yearly_median;
}));

print(collection_yearly_median);

// Create image collection of yearly medians VIIRS
var collection_yearly_viirs = ee.ImageCollection.fromImages(
  years.map(function(y) {
    var yearly_viirs = collection_viirs
```

155

```
        .filter(ee.Filter.calendarRange(y, y, 'year'))
        .reduce(ee.Reducer.median())
        .clip(aoi)
        .rename('VIIRS_median')
        .set('name', 'viirs')
        .set('year', y);
    return yearly_viirs;
}));


print(collection_yearly_viirs);


//////////////////////////////////////////////////
//// VISUALIZE ////
//////////////////////////////////////////////////

Map.addLayer(collection_yearly_greenest.filter(ee.Filter.eq('year',        year)),
imageVisParam, year_string.concat('_greenest'), 0);
Map.addLayer(collection_yearly_median.filter(ee.Filter.eq('year',          year)),
imageVisParam2, year_string.concat('_median'));
Map.addLayer(collection_yearly_viirs.filter(ee.Filter.eq('year',           year)),
imageVisParam3, year_string.concat('_viirs_median'), 0);


Map.addLayer(camps, imageVisParam, 'camps', 0);

//////////////////////////////////////////////////
//// CLUSTERING ////
//////////////////////////////////////////////////

// Create input stack
var classification_stack = (ee.Image.cat([
  collection_yearly_median
  .filter(ee.Filter.eq('year', year))
  .select(['NDVI_median', 'NDBI_median'])
  .toBands()
  .rename(['NDVI', 'NDBI']),
  collection_yearly_median
  .filter(ee.Filter.eq('year', year))
  .select(['B2_median',  'B3_median',  'B4_median',  'B5_median',  'B6_median',
'B7_median', 'B8_median', 'B8A_median', 'B11_median', 'B12_median'])
  .toBands()
  .rename(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12']),
  collection_yearly_viirs
  .filter(ee.Filter.eq('year', year))
  .select('VIIRS_median')
  .toBands()
  .rename('VIIRS')
  ])).set('year', year);
/*
// Create input stack
var classification_stack = (ee.Image.cat([
  collection_yearly_greenest
  .filter(ee.Filter.eq('year', year))
  .select(['NDVI'])
  .toBands()
  .rename(['NDVI']),
  collection_yearly_median
  .filter(ee.Filter.eq('year', year))
  .select(['NDBI_median'])
  .toBands()
  .rename(['NDBI']),
  collection_yearly_greenest
  .filter(ee.Filter.eq('year', year))
  .select(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'])
  .toBands()
  .rename(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12']),
  collection_yearly_viirs
  .filter(ee.Filter.eq('year', year))
  .select('VIIRS_median')
```

```
  .toBands()
  .rename('VIIRS')
])).set('year', year);
*/
print(classification_stack);

// Define which bands for clustering
var bands_clustering = ['NDVI', 'NDBI', 'B2', 'B3', 'B4', 'B5', 'B11', 'B12',
'VIIRS'];

// Make the training dataset
var training = classification_stack.select(bands_clustering).sample({
  region: aoi,
  scale: 10,
  numPixels: 100000
});

// Instantiate the clusterer and train it
var clusterer = ee.Clusterer.wekaKMeans(8).train(training);

// Cluster the input using the trained clusterer
var result = classification_stack.select(bands_clustering).cluster(clusterer);

// Display the clusters with random colors
var palette = ['8b0781', 'FF0000', '888888', '170daa',
               '56d8ff', '53ff3f', '#ffdb58', '#ff3f97'];
Map.addLayer(result, {min: 0, max: 7, palette: palette}, 'clusters', 0);

//////////////////////////////////////////////////
//// ACCURACY ////
//////////////////////////////////////////////////

var result_binary = result.remap([0, 1, 2, 3, 4, 5, 6, 7], [0, 1, 0, 0, 0, 0, 0,
0]).rename('class');
Map.addLayer(result_binary, {min: 0, max: 1, palette: palette}, 'clusters_binary',
0);

var roi_binary = roi.reduceToImage({
    properties: ['class'],
    reducer: ee.Reducer.first()
}).rename('true_class');
Map.addLayer(roi_binary, {min: 0, max: 1, palette: palette}, 'roi_binary', 0);

var validated = result_binary.addBands(roi_binary).sample({
  region: aoi,
  scale: 10,
  projection: 'EPSG:4326',
  //seed: 1
});

var testAccuracy = validated.errorMatrix('true_class', 'class');
print(('Validation error matrix for ').concat(year), testAccuracy);
print(('Validation overall accuracy for ').concat(year), testAccuracy.accuracy());
print(('Validation     producer     accuracy     for     ').concat(year),
testAccuracy.producersAccuracy());
print(('Validation     consumers     accuracy     for     ').concat(year),
testAccuracy.consumersAccuracy());

var difference = (result_binary.multiply(ee.Image(3))).subtract(roi_binary);

//////////////////////////////////////////////////
//// CHARTS ////
//////////////////////////////////////////////////

// Create images to generate chart from
var indices = collection_yearly_median
  .filter(ee.Filter.eq('year', year))
  .select('NDVI_median', 'NDBI_median')
```

```
    .toBands()
    .rename('NDVI', 'NDBI');

var viirs = collection_yearly_viirs
  .filter(ee.Filter.eq('year', year))
  .select('VIIRS_median')
  .toBands()
  .rename('VIIRS');

var bands = collection_yearly_median
  .filter(ee.Filter.eq('year', year))
  .select(bands_median)
  .map(rescale)
  .toBands();

// Define a dictionary that associates band names with values and labels
var indices_info = {
  'NDVI': {v: 1, f: 'NDVI'},
  'NDBI': {v: 2, f: 'NDBI'},
};

// Define client-side custom arrangement of the x-axis
var xPropVals = [];
var xPropLabels = [];
for (var key in indices_info) {
  xPropVals.push(indices_info[key].v);
  xPropLabels.push(indices_info[key]);
}

// Define the indices chart and print it
var chart = ui.Chart.image
              .regions({
                image: indices,
                regions: roi,
                reducer: ee.Reducer.mean(),
                scale: 10,
                seriesProperty: 'type',
                xLabels: xPropVals
              })
              .setChartType('ColumnChart')
              .setOptions({
                title: 'Yearly Median of NDVI and NDBI Values (average over ROIs)',
                hAxis: {
                  title: 'Index',
                  titleTextStyle: {italic: false, bold: true},
                  ticks: xPropLabels
                },
                vAxis: {
                    title: 'Value',
                    titleTextStyle: {italic: false, bold: true},
                    maxValue: 1,
                    minValue: -1
                },
                colors: ['8b0781', 'FF0000'],
              });

print(chart);

// Define the viirs chart and print it
var chart_viirs = ui.Chart.image
                    .byRegion({
                      image: viirs,
                      regions: roi,
                      reducer: ee.Reducer.mean(),
                      scale: 10,
                      xProperty: 'type'
                    })
                    .setSeriesNames([
```

```
                    'VIIRS'
                 ])
                 .setChartType('ColumnChart')
                 .setOptions({
                   title: 'Yearly Median of VIIRS Stray Light Corrected
Nighttime Values (average over ROIs)',
                   hAxis:
                       {title: 'Class', titleTextStyle: {italic: false, bold:
true}},
                   vAxis: {
                     title: 'Value',
                     titleTextStyle: {italic: false, bold: true}
                   },
                   colors: [
                     '555555'
                   ]
                 });

print(chart_viirs);

// Define the spectral chart and print it
var spectralChart = ui.Chart.image.regions(
    bands, roi, ee.Reducer.mean(), 10, 'type', wavelengths)
      .setChartType('ScatterChart')
      .setOptions({
        title: 'Sentinel-2 L1C Spectral Signatures (average over ROIs)',
        hAxis: {title: 'Wavelength (micrometers)'},
        vAxis: {title: 'Reflectance'},
        lineWidth: 1,
        pointSize: 4,
        series: {
          0: {color: '8b0781'},
          2: {color: 'FF0000'},
        }
      });

print(spectralChart);
```

159

Error matrices for the yearly cropland classifications

TRAINING ACCURACY

1_Abadam	2014			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

1_Abadam	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

2_Guzamala	2014			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

2_Guzamala	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

3_Kukawa	2014			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

3_Kukawa	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

4_Marte	2014			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

4_Marte	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

5_Kala_Balge	2014			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

5_Kala_Balge	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0	1	1
Cropland	0	0	1	1
Overall	1			

VALIDATION ACCURACY

1_Abadam	2014			
	Omission	Comission	PA	UA
Non-cropland	0.108696	0.068182	0.891304	0.931818
Cropland	0.6	0.714286	0.4	0.285714
Overall	0.843137			

1_Abadam	2019			
	Omission	Comission	PA	UA
Non-cropland	0.162791	0.162791	0.837209	0.837209
Cropland	0.875	0.875	0.125	0.125
Overall	0.72549			

2_Guzamala	2014			
	Omission	Comission	PA	UA
Non-cropland	0.0625	0.333333	0.9375	0.666667
Cropland	0.652174	0.2	0.347826	0.8
Overall	0.690909			

2_Guzamala	2019			
	Omission	Comission	PA	UA
Non-cropland	0.153846	0.153846	0.846154	0.846154
Cropland	0.375	0.375	0.625	0.625
Overall	0.781818			

3_Kukawa	2014			
	Omission	Comission	PA	UA
Non-cropland	0.113636	0.093023	0.886364	0.906977
Cropland	0.5	0.555556	0.5	0.444444
Overall	0.826923			

3_Kukawa	2019			
	Omission	Comission	PA	UA
Non-cropland	0.12766	0.088889	0.87234	0.911111
Cropland	0.8	0.857143	0.2	0.142857
Overall	0.807692			

4_Marte	2014			
	Omission	Comission	PA	UA
Non-cropland	0.133333	0.315789	0.866667	0.684211
Cropland	0.545455	0.285714	0.454545	0.714286
Overall	0.692308			

4_Marte	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0.038462	1	0.961538
Cropland	1	1	0	0
Overall	0.961538			

5_Kala_Balge	2014			
	Omission	Comission	PA	UA
Non-cropland	0.035714	0.44898	0.964286	0.55102
Cropland	0.611111	0.066667	0.388889	0.933333
Overall	0.640625			

5_Kala_Balge	2019			
	Omission	Comission	PA	UA
Non-cropland	0	0.16129	1	0.83871
Cropland	0.833333	0	0.166667	1
Overall	0.84375			

GEE script for generation of yearly stacks of monthly max NDVI composites (example for Sentinel-2)

```
// Imports
var aoi_NGA = /*table import*/

/////////////////////////////////////////////////////
//// SET PARAMETERS AND ADD AND INSPECT DATA ////
/////////////////////////////////////////////////////

// Center the map over aoi
Map.centerObject(aoi_NGA);
var aoi = aoi_NGA.geometry().bounds();
Map.addLayer(aoi_NGA);

// Define year and months
var year = '2019';
var months = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12];

// Define a function that adds an NDVI band to a Sentinel-2 image
var addNDVI = function(image) {
  var ndvi = image.normalizedDifference(['B8', 'B4']).rename('NDVI');
  return image.addBands(ndvi);
};

/////////////////////////////////////////////////////
//// CREATE STACKS ////
/////////////////////////////////////////////////////

// Define and filter S2 collection based on AOI and map the addNDVI
var collection = ee.ImageCollection('COPERNICUS/S2_SR')
  //.select(bands)
  .filterBounds(aoi_NGA)
  .filterDate(year.concat('-01-01'), year.concat('-12-31'))
  //.filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 50))
  .map(addNDVI);

print(collection);

// Create yearly stacks of monthly maximum NDVI
var yearly_composite = ee.Image.cat(
  months.map(function(m) {
    var monthly_max_ndvi = collection
        .filter(ee.Filter.calendarRange(m, m, 'month'))
        .select('NDVI')
        .reduce(ee.Reducer.max())
        .rename(('vp').concat(m).concat('NDVI_').concat(year));
    return monthly_max_ndvi;
}));

// Add the yearly composites to the map
print (yearly_composite);
Map.addLayer(yearly_composite.clip(aoi_NGA), imageVisParam, year);

/////////////////////////////////////////////////////
//// EXPORT ////
/////////////////////////////////////////////////////

// Export
Export.image.toAsset({
  image: yearly_composite.clip(aoi_NGA),
  description: ('NGA_S2_SR_maxNDVIcomposite_full').concat(year),
  assetId: ('NGA_S2_SR_maxNDVIcomposite_full').concat(year),
  scale: 10,
  region: aoi_NGA,
  maxPixels: 1e10});
```

161

GEE script for stratified multi-temporal supervised classification

```
// Imports
var NGA_L8_2014_ALL =
ee.Image("users/agataelia1991/NGA_L8_SR_maxNDVIcomposite_full2014"),
NGA_S2_2019_ALL =
ee.Image("users/agataelia1991/NGA_S2_SR_maxNDVIcomposite_full2019"),
aoi_ALL = ee.FeatureCollection("users/agataelia1991/AOI_all"),
imageVisParam_2014 =
{"opacity":1,"bands":["vp8NDVI_2014","vp9NDVI_2014","vp10NDVI_2014"],"gamma":1},
imageVisParam_2019 =
{"opacity":1,"bands":["vp8NDVI_2019","vp9NDVI_2019","vp10NDVI_2019"],"gamma":1},
training_areas_agri =
ee.FeatureCollection("users/agataelia1991/TA_LC_250_AOI_ALL_ID_kmeans"),
validation_areas_agri =
ee.FeatureCollection("users/agataelia1991/TA_LC_260_AOI_ALL_ID_10m"),
EMSN_populated_places =
ee.FeatureCollection("users/agataelia1991/Populated_Places");


/////////////////////////////////////////////////////
//// SET PARAMETERS AND ADD AND INSPECT DATA ////
/////////////////////////////////////////////////////

// Define main variables
var bands_2014 = ['vp8NDVI_2014','vp9NDVI_2014','vp10NDVI_2014'];

var bands_2019 = ['vp8NDVI_2019','vp9NDVI_2019','vp10NDVI_2019'];

var ID_list = [1, 2, 3, 4, 5];

// Center the map over aoi
Map.centerObject(aoi_ALL);
Map.addLayer(aoi_ALL, {color: 'F096FF'}, 'aoi', 0);
print(aoi_ALL);

// Add and inspect images
Map.addLayer(NGA_L8_2014_ALL, imageVisParam_2014, '2014_all');
//print (NGA_L8_2014_ALL);
Map.addLayer(NGA_S2_2019_ALL, imageVisParam_2019, '2019_all');
//print (NGA_S2_2019_ALL);

// Add and inspect training areas
Map.addLayer(training_areas_agri, {color: 'F096FF'}, 'training areas', 0);
print (training_areas_agri);

/////////////////////////////////////////////////////
//// CLASSIFICATION 2014 ////
/////////////////////////////////////////////////////

var classfication_2014 = (ee.ImageCollection(
  ID_list.map(function(id){

  // Select aoi and training areas by AOI ID
  var aoi = aoi_ALL.filter(ee.Filter.eq('ID', id));
  var aoi_name = aoi.get('NAME');
  var training_sample = training_areas_agri.filter(ee.Filter.eq('ID_aoi', id));
  var validation_sample = validation_areas_agri.filter(ee.Filter.eq('ID_aoi',
id));

  // Create training dataset
  var training = NGA_L8_2014_ALL.clip(aoi).select(bands_2014).sampleRegions({
  collection: training_sample,
  properties: ['lc_2014'],
  scale: 10
  });

  // Train a classifier
```

```
  var classifier = ee.Classifier.smileRandomForest(100).train({
  features: training,
  classProperty: 'lc_2014',
  inputProperties: bands_2014
  });

  // Get a confusion matrix representing resubstitution accuracy
  var trainAccuracy = classifier.confusionMatrix();
  print(('Resubstitution error matrix for ').concat(id).concat(': '),
trainAccuracy);
  print(('Training overall accuracy for ').concat(id).concat(': '),
trainAccuracy.accuracy());
  print(('Training producer accuracy for ').concat(id).concat(': '),
trainAccuracy.producersAccuracy());
  print(('raining consumers accuracy for ').concat(id).concat(': '),
trainAccuracy.consumersAccuracy());

  // Export matrix
  var trainAccuracy_feature = ee.Feature(null, {matrix: trainAccuracy.array()});
  Export.table.toDrive({
  collection: ee.FeatureCollection(trainAccuracy_feature),
  description: ('trainAccuracy_2014_').concat(id),
  fileFormat: 'CSV'});

  // Run the classification
  var classified =
NGA_L8_2014_ALL.clip(aoi).select(bands_2014).classify(classifier);

  // Sample the validation data
  var validation = NGA_L8_2014_ALL.clip(aoi).select(bands_2014).sampleRegions({
  collection: validation_sample,
  properties: ['lc_2014'],
  scale: 10
  });

  // Classify the validation data
  var validated = validation.classify(classifier);

  // Get a confusion matrix representing expected accuracy
  var testAccuracy = validated.errorMatrix('lc_2014', 'classification');
  print(('Validation error matrix for ').concat(id).concat(': '), testAccuracy);
  print(('Validation overall accuracy for ').concat(id).concat(': '),
testAccuracy.accuracy());
  print(('Validation producer accuracy for ').concat(id).concat(': '),
testAccuracy.producersAccuracy());
  print(('Validation consumers accuracy for ').concat(id).concat(': '),
testAccuracy.consumersAccuracy());

  // Export matrix
  var testAccuracy_feature = ee.Feature(null, {matrix: testAccuracy.array()});
  Export.table.toDrive({
  collection: ee.FeatureCollection(testAccuracy_feature),
  description: ('testAccuracy_2014_').concat(id),
  fileFormat: 'CSV'});

  // Return the output of the classification per AOI
  return (classified);
}))).mosaic();

// Inspect the result
print(classfication_2014);

// Add classified crop mask image
var classification_2_2014 =
classfication_2014.updateMask(classfication_2014.eq(2));
Map.addLayer(classification_2_2014, {min: 2, max: 2, palette: ['#3efd27'],
opacity: 0.6}, 'classification_crop_2014', 0);
```

```
//////////////////////////////////////////////////
//// CLASSIFICATION 2019 ////
//////////////////////////////////////////////////

var classfication_2019 = (ee.ImageCollection(
  ID_list.map(function(id){

  // Select aoi and training areas by AOI ID
  var aoi = aoi_ALL.filter(ee.Filter.eq('ID', id));
  var aoi_name = aoi.get('NAME');
  var training_sample = training_areas_agri.filter(ee.Filter.eq('ID_aoi', id));
  var validation_sample = validation_areas_agri.filter(ee.Filter.eq('ID_aoi',
id));

  // Create training dataset
  var training = NGA_S2_2019_ALL.clip(aoi).select(bands_2019).sampleRegions({
  collection: training_sample,
  properties: ['lc_2019'],
  scale: 10
  });

  // Train a classifier
  var classifier = ee.Classifier.smileRandomForest(100).train({
  features: training,
  classProperty: 'lc_2019',
  inputProperties: bands_2019
  });

  // Get a confusion matrix representing resubstitution accuracy
  var trainAccuracy = classifier.confusionMatrix();
  print(('Resubstitution error matrix for ').concat(id).concat(': '),
trainAccuracy);
  print(('Training overall accuracy for ').concat(id).concat(': '),
trainAccuracy.accuracy());
  print(('Training producer accuracy for ').concat(id).concat(': '),
trainAccuracy.producersAccuracy());
  print(('raining consumers accuracy for ').concat(id).concat(': '),
trainAccuracy.consumersAccuracy());

  // Export matrix
  var trainAccuracy_feature = ee.Feature(null, {matrix: trainAccuracy.array()});
  Export.table.toDrive({
  collection: ee.FeatureCollection(trainAccuracy_feature),
  description: ('trainAccuracy_2019_').concat(id),
  fileFormat: 'CSV'});

  // Run the classification
  var classified =
NGA_S2_2019_ALL.clip(aoi).select(bands_2019).classify(classifier);

  // Sample the validation data
  var validation = NGA_S2_2019_ALL.clip(aoi).select(bands_2019).sampleRegions({
  collection: validation_sample,
  properties: ['lc_2019'],
  scale: 10
  });

  // Classify the validation data
  var validated = validation.classify(classifier);

  // Get a confusion matrix representing expected accuracy
  var testAccuracy = validated.errorMatrix('lc_2019', 'classification');
  print(('Validation error matrix for ').concat(id).concat(': '), testAccuracy);
  print(('Validation overall accuracy for ').concat(id).concat(': '),
testAccuracy.accuracy());
  print(('Validation producer accuracy for ').concat(id).concat(': '),
testAccuracy.producersAccuracy());
```

```
  print(('Validation consumers accuracy for ').concat(id).concat(': '),
testAccuracy.consumersAccuracy());

  // Export matrix
  var testAccuracy_feature = ee.Feature(null, {matrix: testAccuracy.array()});
  Export.table.toDrive({
  collection: ee.FeatureCollection(testAccuracy_feature),
  description: ('testAccuracy_2019_').concat(id),
  fileFormat: 'CSV'});

  // Return the output of the classification per AOI
  return (classified);
}))).mosaic();

// Inspect the result
print(classfication_2019);

// Add classified crop mask image
var classification_2_2019 =
classfication_2019.updateMask(classfication_2019.eq(2));
Map.addLayer(classification_2_2019,  {min: 2, max: 2, palette: ['#3efd27'],
opacity: 0.6}, 'classification_crop_2019', 0);

///////////////////////////////////////////////////
//// CHANGE DETECTION ////
///////////////////////////////////////////////////

var change =
(classfication_2014.multiply(ee.Image(3))).subtract(classfication_2019);
Map.addLayer(change, {min: 1, max: 5, palette: ['#3efd27', '#fdfdfd', '272ef4',
'#f4ef5c', '#fd3819'], opacity: 0.6}, 'change', 0);

///////////////////////////////////////////////////
//// EXPORT ////
///////////////////////////////////////////////////

// Export results of classification
Export.image.toDrive({
  image: classfication_2014,
  description: 'NGA_classified_2014',
  region: aoi_ALL,
  scale: 10,
  fileFormat: 'GeoTIFF'
});

Export.image.toDrive({
  image: classfication_2019,
  description: 'NGA_classified_2019',
  region: aoi_ALL,
  scale: 10,
  fileFormat: 'GeoTIFF'
});

Export.image.toDrive({
  image: change.int(),
  description: 'NGA_change',
  region: aoi_ALL,
  scale: 10,
  fileFormat: 'GeoTIFF',
  maxPixels: 1e10
});

// Add populated places layer
Map.addLayer(EMSN_populated_places, {palette: ['#f4ef5c']}, 'populated places',
0);
```

processSentinel2_ONDA_Uganda_post.py

```python
#Import modules
import os
import subprocess

rootFolder = r'/home/ubuntu/agata/missing_millions/uganda_2020'
lista = ['/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NTH_20200212T113212.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NTJ_20200212T113212.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NTK_20200212T113212.zip',
        '/mnt/catalogue/S2/2A/MSI/LEVEL-
2A/S2MSIL2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NUH_20200212T113212.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NUH_20200209T110919.zip',
        '/mnt/catalogue/S2/2A/MSI/LEVEL-
2A/S2MSIL2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NUJ_20200212T113212.zip',
        '/mnt/catalogue/S2/MSI/LEVEL-
2A/S2MSIL2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NUJ_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NUK_20200212T113212.zip',
        '/mnt/catalogue/S2/MSI/LEVEL-
2A/S2MSIL2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NUK_20200209T110919.zip',
        '/mnt/catalogue/S2/MSI/LEVEL-
2A/S2MSIL2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NVH_20200209T110919.zip',
        '/mnt/catalogue/S2/MSI/LEVEL-
2A/S2MSIL2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NVJ_20200209T110919.zip',
        '/mnt/catalogue/S2/MSI/LEVEL-
2A/S2MSIL2A/2020/02/12/S2A_MSIL2A_20200212T081041_N0214_R078_T36NVK_20200212T113212.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NVK_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/01/20/S2A_MSIL2A_20200120T080241_N0213_R035_T36NWH_20200120T111107.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NWJ_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NWK_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NXH_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NXJ_20200209T110919.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/01/17/S2A_MSIL2A_20200117T075241_N0213_R135_T36NXJ_20200117T105226.zip',
        '/mnt/catalogue/OPTICAL/LEVEL-
2A/2020/02/09/S2A_MSIL2A_20200209T080051_N0214_R035_T36NXK_20200209T110919.zip']

#Search selected bands in MSIL2A products
def searchBands(S2path):
        listBands = []
        for root, dirs, files in os.walk(S2path):
                for filename in files:
                        if "_B02_10m.jp2" in filename:
                                fullPathBand02 = os.path.join(root, filename)
                                listBands.append(fullPathBand02)
                        if "_B03_10m.jp2" in filename:
                                fullPathBand03 = os.path.join(root, filename)
                                listBands.append(fullPathBand03)
                        if "_B04_10m.jp2" in filename:
                                fullPathBand04 = os.path.join(root, filename)
                                listBands.append(fullPathBand04)
        return listBands

#Composite bands
def bandComposite(bandsList, outFile):
        mergeCommand = ["gdal_merge.py", "-o", outFile, "-co",
                        "PHOTOMETRIC=RGB", "-ot", "UInt16", "-separate",
                        bandsList[2], bandsList[1], bandsList[0]]
        subprocess.check_call(mergeCommand)
        return

#Reproject the composites in EPSG:3857
def reproject(inComposite, outWarp):
        reprojectCommand = ["gdalwarp", "-t_srs", "EPSG:3857",
```

166

```
                                    "-srcnodata", "0", inComposite, outWarp]
            subprocess.check_call(reprojectCommand)
            return


#Return list of warped composites
def returnComposites(folderComposites):
            composites = os.listdir(folderComposites)
            listPathsComposites = []
            for composite in composites:
                    compositeFullPath = os.path.join(folderComposites, composite)
                    listPathsComposites.append(compositeFullPath)
            return listPathsComposites


#Traslate tiff into 8 bit tiff
def translate(inWarp, outTranslate):
            translateCommand = ["gdal_translate", "-ot", "Byte", "-strict",
                            "-r", "bilinear", "-scale", "1", "2200", "1", "255",
                            "-colorinterp_1", "red", "-colorinterp_2", "green",
                            "-colorinterp_3", "blue", "-co", "PHOTOMETRIC=RGB", inWarp,
                            outTranslate]
            subprocess.check_call(translateCommand)
            return


#Create virtual mosaic
def buildVRT(outMosaic, listComposites):
            VRTCommand = ["gdalbuildvrt", outMosaic] + listComposites
            subprocess.check_call(VRTCommand)
            return



#Generate tiles
def buildTiles(inMosaic, outTileFolder):
            tileCommand = ["gdal2tiles.py", "-r", "bilinear", "--zoom=5-14",
                        inMosaic, outTileFolder]
            subprocess.check_call(tileCommand)
            return


#Start process
if __name__=="__main__":

            outputFolderComposites = os.path.join(rootFolder, "Composites")
            if not os.path.exists(outputFolderComposites):
                    os.mkdir(outputFolderComposites)

            outputFolderWarped = os.path.join(outputFolderComposites, "Warped")
            if not os.path.exists(outputFolderWarped):
                    os.mkdir(outputFolderWarped)



            outputFolderTranslated = os.path.join(outputFolderWarped, "Translated")
            if not os.path.exists(outputFolderTranslated):
                    os.mkdir(outputFolderTranslated)

            outputFolderTiles = os.path.join(outputFolderTranslated, "Tiles")
            if not os.path.exists(outputFolderTiles):
                    os.mkdir(outputFolderTiles)

            for S2 in lista:
                    bandsPaths = searchBands(S2)
                    print bandsPaths
                    outFilePath = os.path.join(outputFolderComposites,
                                        os.path.basename(S2)[:-4] + ".tif")
                    print outFilePath
                    if not os.path.exists(outFilePath):
                            bandComposite(bandsPaths, outFilePath)

            composites = os.listdir(outputFolderComposites)
            for composite in composites:
                    if composite[-4:] == ".tif":
                            compositeFullPath = os.path.join(outputFolderComposites,
composite)
                            warpPath = os.path.join(outputFolderWarped,
                                            composite[:-4] + "_warped.tif")
                            print warpPath
                            if not os.path.exists(warpPath):
                                    reproject(compositeFullPath, warpPath)
```

```python
warpedComposites = os.listdir(outputFolderWarped)
for warpedComposite in warpedComposites:
        if warpedComposite[-4:] == ".tif":
                warpedCompositeFullPath = os.path.join(outputFolderWarped,
                                                       warpedComposite)
                translatePath = os.path.join(outputFolderTranslated,
                                             warpedComposite[:-11] + "_translated.tif")
                print translatePath
                if not os.path.exists(translatePath):
                        translate(warpedCompositeFullPath, translatePath)

listTranslatedComposites = returnComposites(outputFolderTranslated)
outMosaicPath = os.path.join(outputFolderTranslated, "Mosaic.vrt")
print outMosaicPath
if not os.path.exists(outMosaicPath):
        buildVRT(outMosaicPath, listTranslatedComposites)

buildTiles(outMosaicPath, outputFolderTiles)
```

processSentinel2_ONDA_Uganda_pre_2017.py

```python
#Import modules
import os
import subprocess

lista =['/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NTH_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NTH_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NTJ_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NTJ_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NTK_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NTK_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NUH_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NUH_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/21/S2A_MSIL1C_20171221T080331_N0206_R035_T36NUH_20171221T100353.zip/S2A_MSIL1C_
20171221T080331_N0206_R035_T36NUH_20171221T100353.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NUJ_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NUJ_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NUJ_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NUJ_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/14/S2A_MSIL1C_20171214T081331_N0206_R078_T36NUK_20171214T103532.zip/S2A_MSIL1C_
20171214T081331_N0206_R078_T36NUK_20171214T103532.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NUK_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NUK_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NVH_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NVH_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/16/S2B_MSIL1C_20171216T080319_N0206_R035_T36NVJ_20171216T102645.zip/S2B_MSIL1C_
20171216T080319_N0206_R035_T36NVJ_20171216T102645.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/19/S2B_MSIL1C_20171219T081329_N0206_R078_T36NVK_20171219T105309.zip/S2B_MSIL1C_
20171219T081329_N0206_R078_T36NVK_20171219T105309.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/21/S2A_MSIL1C_20171221T080331_N0206_R035_T36NVK_20171221T100353.zip/S2A_MSIL1C_
20171221T080331_N0206_R035_T36NVK_20171221T100353.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NWH_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NWH_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NWJ_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NWJ_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/21/S2A_MSIL1C_20171221T080331_N0206_R035_T36NWK_20171221T100353.zip/S2A_MSIL1C_
20171221T080331_N0206_R035_T36NWK_20171221T100353.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NXH_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NXH_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/06/S2B_MSIL1C_20171206T080309_N0206_R035_T36NXJ_20171206T102614.zip/S2B_MSIL1C_
20171206T080309_N0206_R035_T36NXJ_20171206T102614.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/13/S2B_MSIL1C_20171213T075319_N0206_R135_T36NXJ_20171213T101202.zip/S2B_MSIL1C_
20171213T075319_N0206_R135_T36NXJ_20171213T101202.SAFE',
        '/mnt/catalogue/OPTICAL/LEVEL-
1C/2017/12/16/S2B_MSIL1C_20171216T080319_N0206_R035_T36NXK_20171216T102645.zip/S2B_MSIL1C_
20171216T080319_N0206_R035_T36NXK_20171216T102645.SAFE']

rootFolder = r'/home/ubuntu/agata/missing_millions/uganda_2017'
inputL1C = r'/home/ubuntu/agata/missing_millions/uganda_2017_L1C'
outputL2A = r'/home/ubuntu/agata/missing_millions/uganda_2017_L2A'

#Copy Sentinel2 to local folder
def copy(L1Cpath, outFolder):
        copyCommand = ["sudo", "cp", "-r", L1Cpath, outFolder]
        subprocess.check_call(copyCommand)
```

```python
                return

#Apply Sen2Cor to L1C data
def atmospheric(L1Cpath, L2Afolder):
        for root, dirs, files in os.walk(L1Cpath):
                for filename in files:
                        if "QI_DATA" in filename:
                                fullQIPath = os.path.join(root, filename)
                                if os.path.isfile(fullQIPath):
                                        deleteQICommand = ["sudo", "rm", fullQIPath]
                                        createQIFolderCommand = ["sudo", "mkdir", root +
"/QI_DATA"]

                                        subprocess.check_call(deleteQICommand)
                                        subprocess.check_call(createQIFolderCommand)
        if os.path.isfile(L1Cpath + "/AUX_DATA"):
                deleteCommand = ["sudo", "rm", L1Cpath + "/AUX_DATA"]
                createFolderCommand = ["sudo", "mkdir", L1Cpath + "/AUX_DATA"]
                subprocess.check_call(deleteCommand)
                subprocess.check_call(createFolderCommand)
        atmosphericCommand = ["sudo", "/home/ubuntu/sen2cor/Sen2Cor-02.08.00-
Linux64/bin/L2A_Process",
                                L1Cpath, "--output_dir", L2Afolder, "--resolution", "10"]
        subprocess.check_call(atmosphericCommand)
        return

#Search selected bands in MSIL2A products
def searchBands(S2path):
        listBands = [1, 2, 3]
        for root, dirs, files in os.walk(S2path):
                for filename in files:
                        if "_B02_10m.jp2" in filename:
                                fullPathBand02 = os.path.join(root, filename)
                                listBands[0] = fullPathBand02
                        if "_B03_10m.jp2" in filename:
                                fullPathBand03 = os.path.join(root, filename)
                                listBands[1] = fullPathBand03
                        if "_B04_10m.jp2" in filename:
                                fullPathBand04 = os.path.join(root, filename)
                                listBands[2] = fullPathBand04
        return listBands

#Composite bands
def bandComposite(bandsList, outFile):
        mergeCommand = ["gdal_merge.py", "-o", outFile, "-co",
                        "PHOTOMETRIC=RGB", "-ot", "UInt16", "-separate",
                        bandsList[2], bandsList[1], bandsList[0]]
        subprocess.check_call(mergeCommand)
        return

#Reproject the composites in EPSG:3857
def reproject(inComposite, outWarp):
        reprojectCommand = ["gdalwarp", "-t_srs", "EPSG:3857",
                                "-srcnodata", "0", inComposite, outWarp]
        subprocess.check_call(reprojectCommand)
        return

#Return list of warped composites
def returnComposites(folderComposites):
        composites = os.listdir(folderComposites)
        listPathsComposites = []
        for composite in composites:
                compositeFullPath = os.path.join(folderComposites, composite)
                listPathsComposites.append(compositeFullPath)
        return listPathsComposites

#Traslate tiff into 8 bit tiff
def translate(inWarp, outTranslate):
        translateCommand = ["gdal_translate", "-ot", "Byte", "-strict",
                                "-r", "bilinear", "-scale", "1", "2200", "1", "255",
                                "-colorinterp_1", "red", "-colorinterp_2", "green",
                                "-colorinterp_3", "blue", "-co", "PHOTOMETRIC=RGB", inWarp,
                                outTranslate]
        subprocess.check_call(translateCommand)
        return

#Create virtual mosaic
```

```python
def buildVRT(outMosaic, listComposites):
        VRTCommand = ["gdalbuildvrt", outMosaic] + listComposites
        subprocess.check_call(VRTCommand)
        return


#Generate tiles
def buildTiles(inMosaic, outTileFolder):
        tileCommand = ["gdal2tiles.py", "-r", "bilinear", "--zoom=5-14",
                       inMosaic, outTileFolder]
        subprocess.check_call(tileCommand)
        return

#Start process
if __name__=="__main__":

        outputFolderComposites = os.path.join(rootFolder, "Composites")
        if not os.path.exists(outputFolderComposites):
                os.mkdir(outputFolderComposites)

        outputFolderWarped = os.path.join(outputFolderComposites, "Warped")
        if not os.path.exists(outputFolderWarped):
                os.mkdir(outputFolderWarped)


        outputFolderTranslated = os.path.join(outputFolderWarped, "Translated")
        if not os.path.exists(outputFolderTranslated):
                os.mkdir(outputFolderTranslated)

        outputFolderTiles = os.path.join(outputFolderTranslated, "Tiles")
        if not os.path.exists(outputFolderTiles):
                os.mkdir(outputFolderTiles)

        for originalL1C in lista:
                L1Cbasename = os.path.basename(originalL1C)
                L1CoutPath = os.path.join(inputL1C, L1Cbasename)
                if not os.path.exists(L1CoutPath):
                        copy(originalL1C, inputL1C)

        L1C_list = os.listdir(inputL1C)
        for L1C in L1C_list:
                L1Cpath = os.path.join(inputL1C, L1C)
                atmospheric(L1Cpath, outputL2A)

        L2A_list = os.listdir(outputL2A)
        for L2A in L2A_list:
                S2 = os.path.join(outputL2A, L2A)
                bandsPaths = searchBands(S2)
                print bandsPaths
                outFilePath = os.path.join(outputFolderComposites,
                                        os.path.basename(S2)[:-4] + ".tif")
                print outFilePath
                if not os.path.exists(outFilePath):
                        bandComposite(bandsPaths, outFilePath)

        composites = os.listdir(outputFolderComposites)
        for composite in composites:
                if composite[-4:] == ".tif":
                        compositeFullPath = os.path.join(outputFolderComposites,
composite)
                        warpPath = os.path.join(outputFolderWarped,
                                                composite[:-4] + "_warped.tif")
                        print warpPath
                        if not os.path.exists(warpPath):
                                reproject(compositeFullPath, warpPath)

        warpedComposites = os.listdir(outputFolderWarped)
        for warpedComposite in warpedComposites:
                if warpedComposite[-4:] == ".tif":
                        warpedCompositeFullPath = os.path.join(outputFolderWarped,
                                                        warpedComposite)
                        translatePath = os.path.join(outputFolderTranslated,
                                                warpedComposite[:-11] + "_translated.tif")
                        print translatePath
                        if not os.path.exists(translatePath):
                                translate(warpedCompositeFullPath, translatePath)
```

```
listTranslatedComposites = returnComposites(outputFolderTranslated)
outMosaicPath = os.path.join(outputFolderTranslated, "Mosaic.vrt")
print outMosaicPath
if not os.path.exists(outMosaicPath):
        buildVRT(outMosaicPath, listTranslatedComposites)

buildTiles(outMosaicPath, outputFolderTiles)
```