Global Earth Monitor

Deliverable 5.7

Demonstration

PREPARATION SLIP

	Name	Organisation	Date
Prepared by	Francesco Perfetto, Ashish Dhiman	TomTom	24/03/2023
Prepared by	Christoph Ramshorn, Nicoletta Addimando	meteoblue	22/09/2023
Prepared by	Michele Lazzarini, Adrian Luna, Omar Barrilero	SatCen	22/09/2023
Prepared by	Matej Batič, Nejc Vesel	Sinergise	22/09/2023
Reviewed by	Marco Körner, Francesco Perfetto, Matej Batič	TUM, TomTom, Sinergise	6/10/2023
Approved for submission by	Mariza Pertovt	Sinergise	10/10/2023

EXECUTIVE SUMMARY

This deliverable provides an overview of the GEM demonstration phase. During this phase, all 5 GEM use cases were run on the selected DEMO domains. For each use case scope area, the ML model, input data and prediction results are provided.

A large-scale, cost-effective continuous monitoring service is also presented in this deliverable. The Continuous Monitoring service aimed to explore the use of the GEM framework for continuous monitoring, with a specific focus on water monitoring in the Sahel region as a showcase.

The results collected during the demonstration phase are also visible via dedicated viewer and/or client applications. In addition, 4 demonstrators from meteoblue are presented.

The report also provides a comprehensive list of the public collections shared by the GEM Consortium that represent the results of the demonstrations.

Contractual Delivery Date	31.08.2023
Actual Delivery Date	10.10.2023
Type of delivery	Demonstrator
Dissemination level:	Public

Table of Contents

List of Figures

List of Tables

1 Introduction

As the name of this deliverable suggest, the scope is to provide an outline of the GEM Demonstration phase. During this phase, all 5 GEM use cases were continuously running on the selected DEMO areas. The document provides an overview of the results collected in the selected DEMO areas, also visible via dedicated viewer and/or client applications, preliminary conclusions, list of public collections representing results from the demonstrations, and a dedicate section describing the Continuous Monitoring Service implemented by Sinergise. The D5.8 Validation Report will summarize all the findings from the Demonstration period.

As previously mentioned, the GEM project has one generic use case, namely, the Land Cover Continuous Monitoring Service (LC CMS) and 4 specific ones which are its derivatives i.e., the Map Making, Conflict Pre-Warning, Crop-Detection and Built-Up area identification use case.

In Built-Up use case we have used GEM platform (eo-learn and eo-grow) to establish a process that can identify new urban areas at large scale on quarterly or even monthly basis. We experimented on the area of Slovenia and France, where high quality reference data provided the means to understand the issues of using low resolution inputs. We have relied on ground truth data provided by TomTom for the purposes of GEM project and based our experiments and analysis on expertise from TomTom (LULC classification) and TUM (ML models) gathered through bi-lateral collaborations and meetings. We have also created the 120 m resolution global cloudless mosaic Analysis Ready Data cube specifically with cost-optimised large-scale monitoring in mind.

We developed a process, which can (on an ongoing basis) produce information about new built-up areas on Sentinel-2 120 m mosaic. To detect built-up areas, we tried several approaches: classifying all the land cover classes within GEM taxonomy, binary classification and finally predicting the fraction of pixel that is considered built-up via regression.

The process heavily relies on eo-learn, which is upgraded with the eo-grow framework for scalability. Computations can be done on remote clusters and scalability was tested by producing the results for the whole of Africa. Thanks to the 120 m resolution the process is fast and cheap, making it a perfect demonstrator of how to run large scale EO data pipelines and workflows.

The goal of the LC-CMS use case is to perform fully automated, efficient, and repeatable global Land Cover mapping for small and mid-scale features. It is a generic GEM use case which shows global Continuous Monitoring capabilities of the GEM project. During the Construction phase, we first used an area of interest (AOI) covering France to performs experiments, and predictions at different resolutions, namely at 120 m, 60 m and 20 m by using quarterly composites for the year 2020. As the predictions at a resolution of 120 m are cheaper to produce, we used Sentinel-2 at 120 m to perform change detection by tracking the changes happening on the land surface for all the land classes over time which can be used for continuous land monitoring and evolution. We finally validated and demonstrated the results over the demonstration Sahara region of Africa.

The objective of the Map Making use case is to create "map-ready" features derived from some upstream process and enrich TomTom's core map feature set by ingesting those features. To achieve this objective, a dedicated pipeline based on LC-CMS pipeline was built, capable of detecting water features. The pipeline was used to perform water features/extraction on several areas, based on the internal TomTom use case. The

results have been tested in the region of Myanmar and demonstrated in several country-size region, included parts of Chile.

The Crop Identification use case demo allows users to inspect the best result we could achieve by combining crop and weather data. Learnings during the projects had us supersede this use case with weather and climate services, use cases, and demonstrators as shown in Table 1.1.

Table 1.1 Services and use cases superseding the Crop Identification use case

Demonstration Application	Enabling Service	Deliverable Section	Work Package
Crop Identification	*ML workflows*	4.4	WP5
City Heat Maps on meteoblue.com website	*Hyper-resolution (10 m) live temperature data in built-up areas*	4.5	WP2, WP5, WP6
Jupyter Notebook	*Climate variables and climate indices on grids*	4.6	WP2, WP6
Measurement Interface (meteoblue internal)	*Measurement data management*	4.7	WP2

The goal of the Conflict Pre-Warning Map use case is to merge multiple data sources to bring relevant information for addressing climate-security issues. The demonstration focuses on supporting the geospatial intelligence production across different phases. In order to do that, the demo supports three steps. In a first step, wide subcontinental (Sahel area) automated continuous monitoring at coarse resolution is implemented. When more details are needed (e.g., request from decision makers), a second step provides semi-automatic approaches to provide GEOINT analysts with standardized sets of data (across different dimensions) and ad-hoc procedures or analyses are used to gain situational awareness at medium-high resolution. Niger is monitored at this level in the demonstration. Finally, on a third step, finer resolution data is used to perform ad-hoc analyses, which support the generation of change maps, impact assessments or contingency plans. Three small AOIs are selected for the demo.

The rest of this deliverable report is structured as follows:

Section 2 for each GEM use case, the Demonstration section provides a description of the experiments performed in the DEMO area.

Section 3 describes how to employ the Continuous Monitoring service through the GEM Framework.

Section 4 shows the dedicated clients/demonstrator developed to visualize, compare, and validate results for each GEM use case.

Section 5 lists all publicly available Collection used in the Demonstration phase.

In **Section 6** we conclude the deliverable with an outline of the preliminary results.

2 Demonstration Use Case

The current section provides an overview of the scope area used in the Demonstration phase for each use case.

2.1 LC-CMS

The below section provides a summary of the experiments performed for the LC-CMS on the selected DEMO area, the full contents is available in section 7 of D5.3 LC-CMS deliverable.

2.1.1 Overview

The goal of the LC-CMS use case is to perform fully automated, efficient, and repeatable global Land Cover (LC) mapping for small and mid-scale features. It is a generic GEM use case which shows global Continuous Monitoring capabilities of the GEM project. LC-CMS use case is of special interest to TomTom as it provides a more efficient, scalable, and maintainable alternative to the previous TomTom LC production pipeline (Earth Cover Engine) based on the capabilities of eo-learn.

To achieve this objective, an end-to-end machine learning based pipeline was built, capable of detecting Level 2 land classes of GEM Taxonomy using mid and low-resolution Sentinel-2 spectral bands. The pipeline was developed using eo-grow and eo-learn frameworks (GEM processing framework) developed by Sinergise. eo-grow allows to easily scale-up and distribute the workload. The LC-CMS pipeline is described in detail within the section 4 of D5.3 LC-CMS deliverable.

The pipeline was used to perform various experiments on the Sentinel-2 bands at different spatial resolutions, ranging from 120 m to 10 m per pixel. Following the lessons learned from TomTom's Earth Cover Engine, we used a LGBM Random Forest Classifier model to perform land pixel classification using a combination of temporal Sentinel 2 bands as input features. The training data was obtained by manually labelling pixels, for different Level 2 classes in GEM Taxonomy, for the region of France for the year of 2020.

We first used an area of interest (AOI) covering France and performed predictions at 120 m, 60 m and 20 m by using quarterly composites for the year 2020. As the predictions at a resolution of 120 m are cheaper to produce, we used Sentinel-2 at 120 m to perform change detection by tracking the changes happening on the land surface for all the land classes over time which can be used for continuous land monitoring and evolution. Together with the Drill Down mechanism the process becomes very cost efficient, globally repeatable, and scalable as it keeps track of only the changes over time at lower resolution and performs predictions on higher resolution imagery only if the change is worth following. We looked at different approaches of change detection and demonstrate the results over France and Sahara region of Africa.

2.1.2 Scope Area

As part of demonstration, we choose the sub-Saharan region in Africa, as shown in Figure 1.

Figure 1: Pilot AOIs and the experiment AOIs

2.1.3 Model Type & Parameters

The LC-CMS pipeline described in section 4 LC-CMS Architecture of D5.3 LC-CMS deliverable was used to perform all the experiments. For each year we used Sentinel-2 L2A data at 120m resolution, and TomTom Ground-Truth reference data to train a model which was predicting pixel-based occurrence of LC classes using LGBM Random Forest.

We first decided to use a single model for the whole AOI of Africa. However, we found that using a single model for the entire area resulted in poor model performance and visible artifacts in the predictions. Also, there were a lot of misclassifications like sea pixels classified as bare soil. Because of the sheer size of the AOI, there's a lot of diversity in the landscapes and the biomes and the features across landscapes are not consistent. For example, two same land class pixels belonging to regions that are far apart in different biomes might have very different values. This limits the model's ability to generalize well. Additionally, there are many satellite swath artifacts that are clearly visible in the overlapping regions of the satellite orbits of S2.

To improve the accuracy of our predictions, we then used multiple models, each trained on a smaller AOI with more consistent characteristics. This allowed the models to generalize better and produce more accurate predictions. There are some new artifacts like sharp change in the prediction class at the region boundaries, but such artifacts can be easily mitigated by capturing more training data at region boundaries.

The Figure 2 shows the pilot region in Africa split in four different sub-regions i.e., AOIs.

Figure 2: Demonstrator AOIs

A comparison of 2019 predictions using a single AOI and four smaller AOIs is shown in Figure 3. The results are more consistent and appear to be more accurate when compared with EC satellite imagery, and the satellite swath artifacts are also greatly reduced.

Figure 3 : Overall predictions for LC-CMS pipeline for 2020.

2.1.4 Input Data

For the LC-CMS pipeline we mainly used Sentinel-2 bands at different spatial resolutions, ranging from 120 m to 10 m per pixel, and the TomTom Ground Truth data. We also performed predictions with and without weather data, for which we concluded that the weather data doesn't contribute to improving predictions at 120m, rather it creates some unexpected artifacts and misclassifications in many regions.

In order to continually run model at low resolution at global scale, Data Cubes (DCs) have been generated by Sinergise. Table 2.1 shows the description of two static DCs that were created using Data Cube processing engine (see deliverable D3.3 – GEM processing Framework for details about Data Cube engine) and eo-grow for the Pilot Region in Africa, one for Sentinel-2 Level-2A data and the second for weather data.

The weather data was retrieved from meteoblue services and contains 9 meteorological parameters: CLOUD_COVER_TOTAL, PRESSURE, SUN_DURATION, VAPOR_PRESSURE_DEFICIT, WIND, HUMIDITY, TEMPERATURE, SOIL_MOISTURE and PRECIPITATION. The data was aggregated on monthly basis with 25th quantile for Sentinel-2 data and mean aggregation for the weather data. The creation of the data cube made it very easy to iterate over the experiments for various use cases since the data aggregation was already completed.

Both DCs are now available through Sentinel-Hub services, and a Jupyter notebook illustrating how to use the static data cubes has been added to eo-learn-examples repository: https://github.com/sentinel-hub/eo-learn-examples/blob/main/GEM-data/gem-collections.ipynb

Table 2.1 Data cubes available in collections and used for experiments and demonstration

Data Source	Bands	Temporal Agg.	Temporal Agg.	Spatial Res.	Resampling	Availability
Sentinel-2	B01, B02, B03, B04, B05, B06, B07, B08, B8A, B09, B11, B12	Monthly	25th Quartile	120 m	Bilinear	2019, 2020, 2021
meteoblue weather data	CLOUD_COVER_TOTAL, PRESSURE, SUN_DURATION, VAPOR_PRESSURE_DEFICIT, WIND, HUMIDITY, TEMPERATURE, SOIL_MOISTURE, PRECIPITATION	Monthly	Mean		Bilinear	

2.1.5 Prediction Results

The purpose of the demonstration was to showcase the capabilities of the LC-CMS (Land Cover Classification Modelling System) pipeline to run on a global scale in a repeated manner. To achieve this, the LC-CMS pipeline was run for three consecutive years, 2019, 2020, and 2021, at 120 m resolution.

We also performed a different Change Detection approach by comparing the predicted results at lower resolution of 120 m and demonstrated for the pilot region. The initial results show that it is feasible to detect changes using this approach however, there's a lot of noise in the detected changes. We have identified several areas for improvement that we can pursue in future work.

The final results of LC Prediction and Change Detection using diff of predictions for years 2019, 2020, and 2021 are available on the LC-CMS Map Viewer available at:
http://globalearthmonitor.eu/sites/default/files/LC_CMS/index.html

2.1.5.1 LC-CMS Prediction

Initial results immediately show that the quality of predictions was reasonable, particularly at such resolution. Figure 4 shows model validation results for all the AOIs for the year 2019. The predictions can capture the global landscape trends in the sub-regions. The model particularly struggled to classify forests correctly and often misclassifying them as bare soil or grass. Overall, the results indicate that the model performs well on distinct classes like water, built-up etc. but struggles with classes that are easily confused with each other like forest, grass, bare soil, farmland. Overall, the mean accuracy of the results was around 0.81, 0.806 and 0.76 for 2019, 2020, 2021 respectively. A summary of the results can be found in Table 2.2.

The results of predictions can be visualized by selecting the layers **LCMS Pilot 120m 2019, LCMS Pilot 120m 2020, LCMS Pilot 120m 2021** in the demonstrator app mentioned in section 4.1.

Table 2.2 Summary of results for LC-CMS models for years 2019, 2020, 2021

	AOI 1			AOI 3			AOI 2 + AOI 4		
Year	2019	2020	2021	2019	2020	2021	2019	2020	2021
Mean Accuracy	0.86	0.89	0.84	0.8	0.82	0.77	0.71	0.71	0.69
F1 Macro	0.44	0.46	0.42	0.53	0.55	0.53	0.56	0.57	0.53
F1 Weighed	0.89	0.91	0.88	0.83	0.85	0.81	0.72	0.72	0.69

AOI-1, 2019 @120m

	precision	recall	f1-score	support
forest	0.07	0.59	0.13	138
farmland	0.05	0.50	0.10	779
bare soil	0.99	0.85	0.92	122033
sand	0.59	0.97	0.73	11097
built-up	0.09	0.43	0.15	362
permanent water	0.48	0.88	0.62	443
intermittent water	0.33	0.76	0.46	387
accuracy			0.86	135239
macro avg	0.37	0.71	0.44	135239
weighted avg	0.95	0.86	0.89	135239

AOI-3, 2019 @120m

	precision	recall	f1-score	support
forest	0.94	0.91	0.92	25921
grass	0.04	0.23	0.06	382
farmland	0.40	0.69	0.51	4760
bare soil	0.97	0.73	0.84	33664
sand	0.11	0.78	0.19	178
built-up	0.22	0.79	0.35	878
permanent water	0.92	0.90	0.91	1948
intermittent water	0.29	0.75	0.42	296
accuracy			0.80	68027
macro avg	0.49	0.72	0.53	68027
weighted avg	0.90	0.80	0.83	68027

AOI-2 + AOI-4, 2019 @120m

	precision	recall	f1-score	support
forest	0.70	0.69	0.70	46966
grass	0.71	0.89	0.79	67755
farmland	0.21	0.70	0.33	2612
bare soil	0.90	0.56	0.69	80399
sand	0.61	0.91	0.73	6618
built-up	0.11	0.64	0.19	715
permanent water	0.62	0.93	0.74	933
intermittent water	0.22	0.71	0.33	413
accuracy			0.71	206411
macro avg	0.51	0.75	0.56	206411
weighted avg	0.77	0.71	0.72	206411

Figure 4: Model validation results for 2019 at 120m for all the AOIs

2.1.5.2 Change Detection

The section 6 of D5.3 LC-CMS describes experiments performed on Change Detection. We performed a basic change detection by comparing the predicted land cover classes over the years at lower resolution of 120 m and demonstrated for the DEMO area. This allowed us to detect changes in land cover for every pixel.

The initial results show that it is feasible to detect changes using this approach however, there's a lot of noise in the detected changes. We have identified several areas for improvement that we can pursue in future work. For example, we can try to filter the noise in the change detection, we can try to increase the quality of predictions of the LC-CMS model, experiment with different kernel sizes for sieving before change detection, create class-specific models for change detection, and investigate the use of higher-resolution imagery for improved quality of predictions.

Table 2.3 shows some examples of the changes detected by the model, overlaid on reference imagery from satellite imagery for the same time period along with the location and description of the change. It should be noted that the satellite imagery is used as a reference and is not considered to be ground truth. This is because it only shows a snapshot of the year, while our model uses S2 imagery which also captures seasonality.
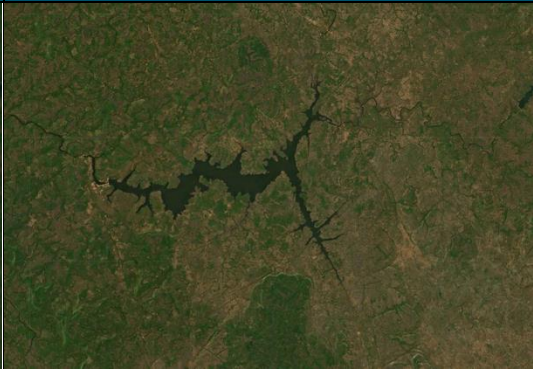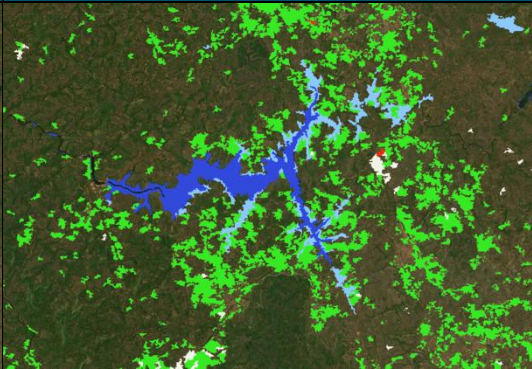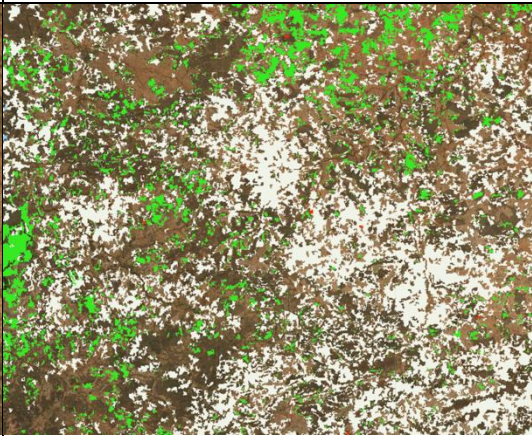
The results of change can be visualized by selecting the layers **LCMS Change 2019-2020, LCMS Change 2020-2021, LCMS Change 2019-2021** in the demonstrator app mentioned in section 4.1.
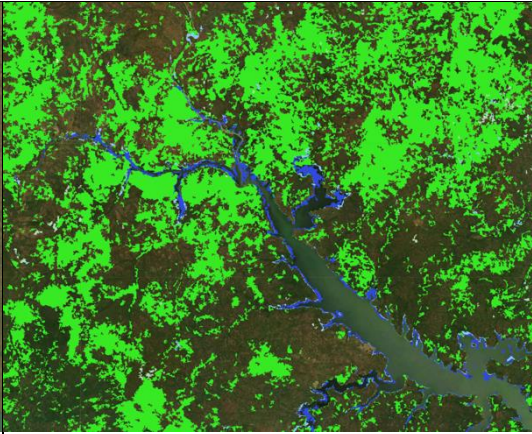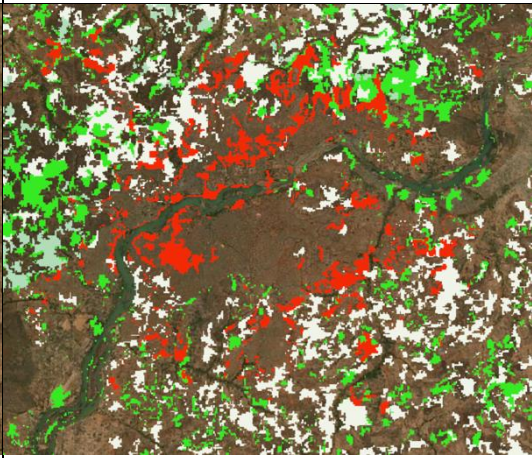
2.1.6 Costs
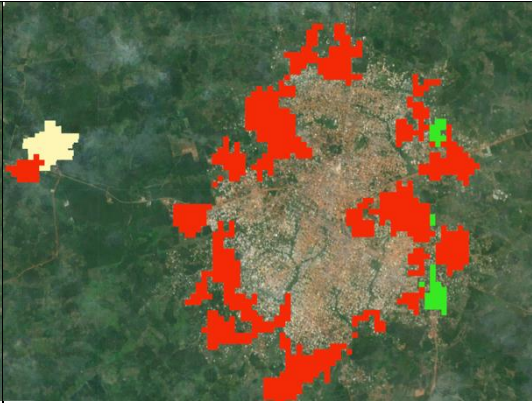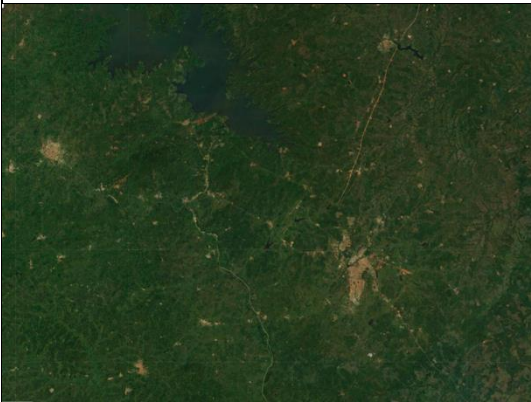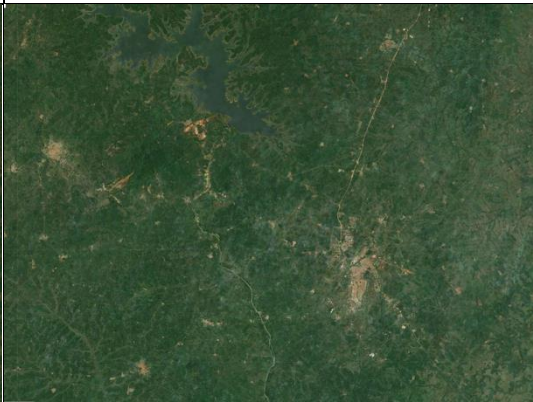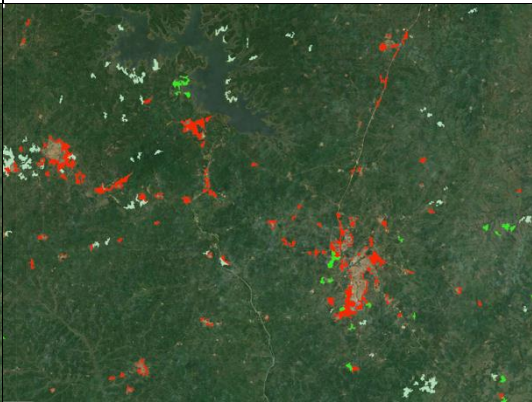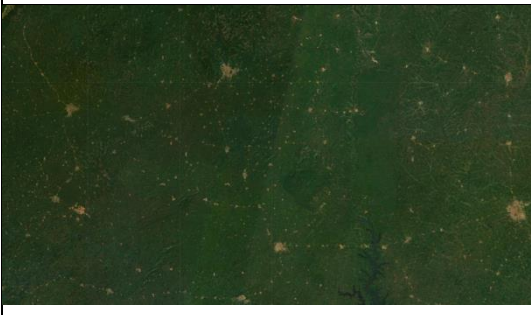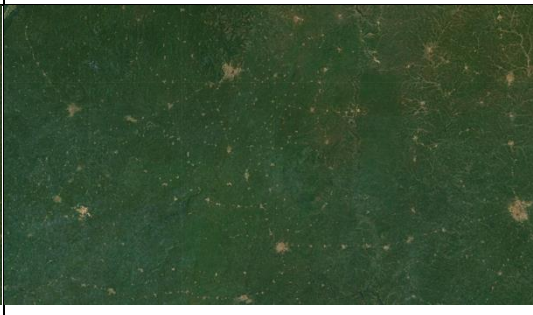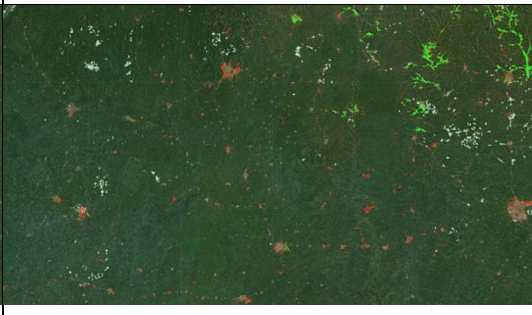
The pipeline was run on four sub-regions in the pilot region for the years **2019, 2020, and 2021** for the static DCs. Thanks to the availability of static DCs, which allowed quick iteration over results. The pipeline was run on AWS Cloud using 5 *m5.xlarge* instances with distributed workload via ray clusters, and the whole process took less than half a day to run. The total AWS cost for processing and saving the data for 3 years of data on Amazon S3 bucket (excluding Sentinel-Hub costs), including the Ground-truth data on AWS RDS instance, was less than 600,00 $.
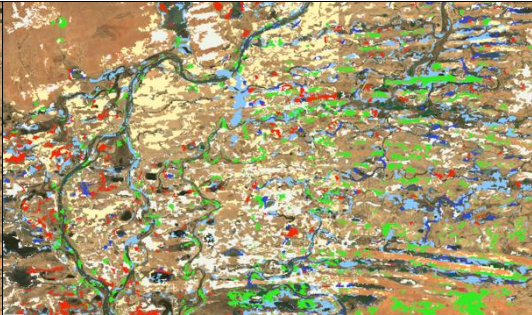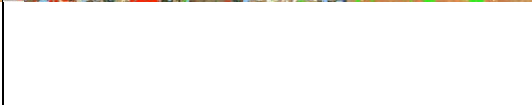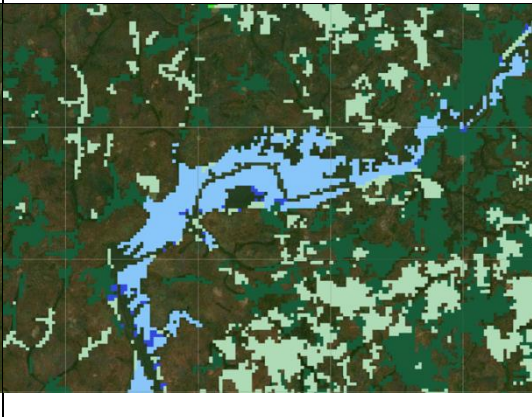
The pilot area is roughly 8.5 million squares kilometres, which is around 7% of global land coverage. If we extrapolate the cost to the entire global land coverage, the cost for three years would be around 8571,00 $. Running LC-CMS globally for one year at 120m resolution should cost around $3000,00. It is to be noted that these are rough estimates based on the modelling process used, and for an input resolution of 120m. The cost significantly varies depending on the modelling technique and the input data resolution used (e.g., using higher resolution and using deep learning models will increase the costs). Additionally, these estimates do not include the manual capturing training data, which it is a significant cost.

Table 2.3 Examples of Change detection tracking the difference between of predictions in 2019, 2020, 2020

Satellite Imagery 2019	Satellite Imagery 2021	Change Detection (Diff of predictions)	Description of detected change
			Reservoir Souapiti in Guinea can be seen to increase in size clearly in the satellite imagery.
			Around **Nioro & Kayes region** in Mali region, a lot of bare soil and vegetation is detected which is in correspondence with the changes in imagery.

Northern basin of **Lake Volta** in Ghana seems to have increased in basin and there is a lot of farmland detected, some of it is noise.

Built up, bare soil & farmland detected in **Bamako City** in Mali are visible in imagery; city's annual population is growing at ~ 3.5% since 2020

Divo city in Cote D'Ivoire shows built up increase and is detected in imagery.

Yamoussoukro and neighbouring areas in Cote D'Ivoire show a lot of built-up growth in imagery and is detected by LC-CMS.

Western parts of Cote D'Ivoire shows built-up, and farmland increase and is consistent with the imagery

Near **Mopti** town in Mali, dried up waterways results in detection of farmland, grass, change from permanent to intermittent water with some noise.

Lake Mbakaou showing change of permanent water changing to intermittent and forest between 2019 & 2020.

2.2 Map Making Demonstration

The below section provides a summary of the experiments performed for the Map Making use case on the selected DEMO area.

2.2.1 Overview

The objective of the Map Making use case is to create "map-ready" features derived from some upstream process and enrich TomTom's core map feature set by ingesting those features. In addition to generating "map-ready" land cover (LC) features for updating and visually improving TomTom's Core Map, extraction performed at 10 m resolution—although sometimes insufficient for creating features at the highest zoom levels—can still be utilized to identify locations where existing features may have changed and maps should be updated. Upon detecting such alterations, these leads can be supplied to editors for updating the product, accordingly, utilizing GEM's drill-down capabilities.

To achieve this objective, a dedicated pipeline based on LC-CMS pipeline was built, capable of detecting water features. The pipeline was used to perform water features/extraction on several areas, based on the internal TomTom use case. The results have been demonstrated in the region of Myanmar and parts of Chile. Myanmar was also used for experimentation, finalizing input bands, and evaluating post-processing and vectorization techniques. So, the viewer may have more layers for Myanmar than Chile Subregion.

2.2.2 Scope Area

As part of demonstration phase, several Area of Interest (AOI) e.g., countries, have been chosen. The table below provides a summary of all AOI and parameters linked to that.

Table 2.4 Demonstration Area

AOI	Country Area (Km2)	AOI Area (Km2)	Type	Resolution (m)	Prediction Year
Chile	758.556	2.269.410	Demonstration	10	2019, 2019, 2019, 2019, 2019
Taiwan	36.785	417.111	Demonstration	10	2019
South Korea	100.672	578.760	Demonstration	10	2019
Uruguay	193.200	600.320	Demonstration	10	2019
Eastern Turkey	337.689	337.689	Demonstration	10	2019-2022
Iraq & Siria	406.128	384.641	Demonstration	10, 60, 120	2019-2022

Myanmar	67.958	1.181.894	Testing	10, 20, 60	2019-2020

The Figure 2: Demonstrator AOIs, shows on map the area selected for the demonstration. Preliminary experiments have been performed, as part of D5.3, for Myanmar and small areas in Chile. The experiments allowed to select the model type and parameters for the final demonstration.

Figure 5: Overview of the Map Making AOI.

2.2.3 Model Type & Parameters

A part of D5.3 section 4, several tests have been performed to select the model type and parameters. Based on the analysis, **we concluded that the Gradient Boosted Decision Trees (GBDT) model with balanced class weight** performs better than the Random Forest (RF) model at predicting water.

In addition, the following parameters were used:

- Metric = multi-log loss

- Class weight = balanced

- The following three parameters are aimed at speeding up model training and avoiding overfitting.

   - Bragging frequency = 1

   - Bagging fraction = 0.625

   - Feature fraction = 0.625

We then conducted feature attribution analysis using only GBDT. For the experiments the following **input data** was considered while selecting and engineering features for Water detection:

- **Sentinel 2 L2A 12 Bands**[1]: B01, B02, B03, B04, B05, B06, B07, B08, B8A, B11 & B12 downloaded using BatchAPI

- **Resolution**: 10m

- **Duration**: 1 Year (2020)

- **Aggregation**: Bi-monthly using 25th Quartile resulting in 6 temporal measurements

ML Model: LightGBM GBDT Classifier for pixel-wise classification.

Performance Measure: We primarily use **F1 score** and **ROC AUC** for choosing the model.

Test Region: Scope region of Myanmar shown in Figure 1 was used for this exercise.

Table 2.5 highlights the final seven remaining bands, including the NDWI index, that we retained for the Water Prediction task. This resulted in a total of 42 features, as opposed to the initial 72 features.

Table 2.5 Final selected Sentinel-2 Bands for Training using SHAPley analysis.

Band	Native Resolution	Description
B01	60m	Ultra-Blue & important features according to SHAP
B04	10m	Red is helpful in detecting longer wavelength in visible spectrum
B05	20m	NIR & important feature according to SHAP
B09	60m	Water vapour, important according to SHAP
B11	20m	SWIR & important feature according to SHAP

---

[1] Sentinel2 L2A Bands in SentinelHub: https://docs.sentinel-hub.com/api/latest/data/sentinel-2-l2a/

B12	20m	SWIR & important feature according to SHAP
B02	10m	Blue weight to avoid loss of native resolution at 10m
NDWI	10m	Normalized Difference Water Index Calculated as (B03 − B08)/ (B03 + B08)

2.2.4 Input Data

For the Map Making pipeline, we mainly used Sentinel-2 bands at different spatial resolutions, ranging from 120 m to 10 m per pixel, and the TomTom Ground Truth data. In summary:

- Sentinel-2 bands at 10 m resolution were used for training.

- Data used varies for each AOI, the selected years are shown in Table 2.4. The measurements for the whole year were aggregated bi-monthly, resulting in a total of 6 temporal measurements.

- Only a select number of bands and indices, as mentioned in Table 2.5 derived after feature engineering were used for training the model.

2.2.5 Prediction Results
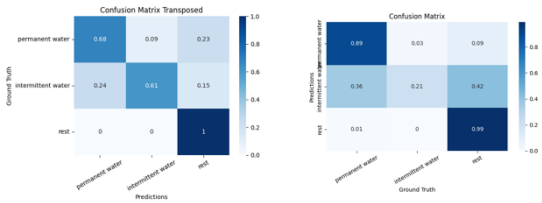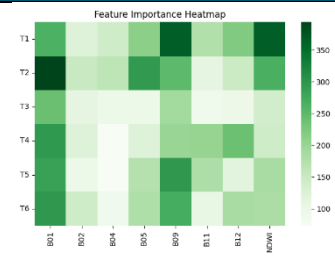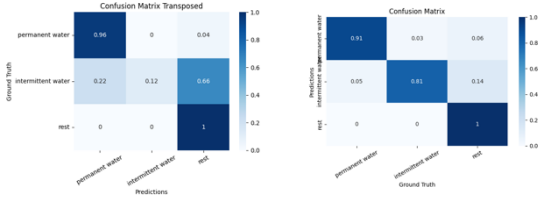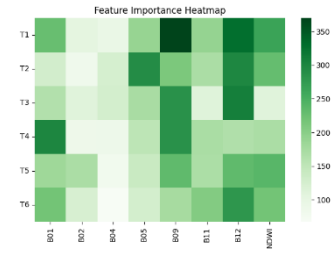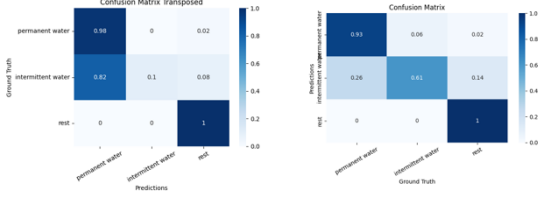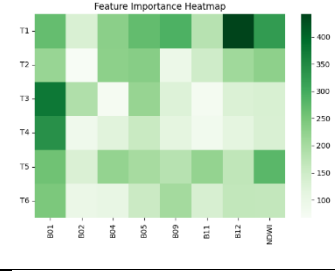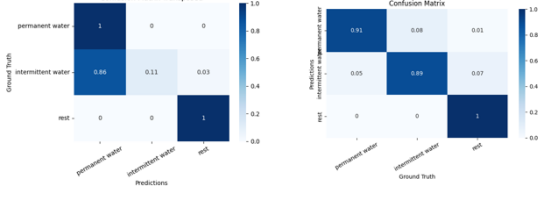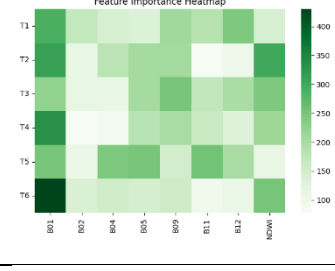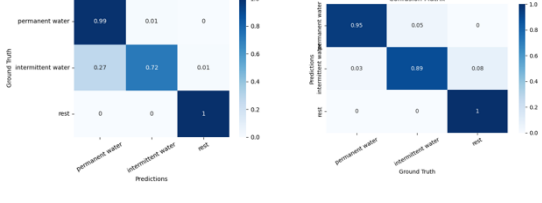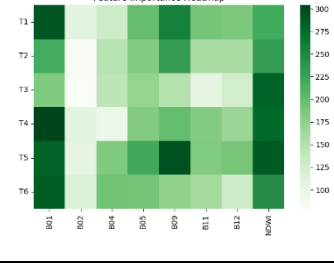
2.2.5.1 Water Predictions
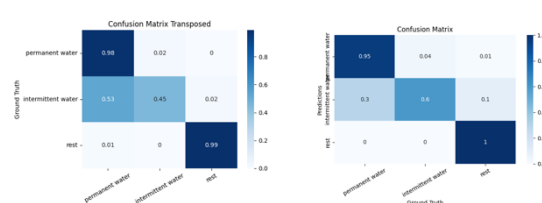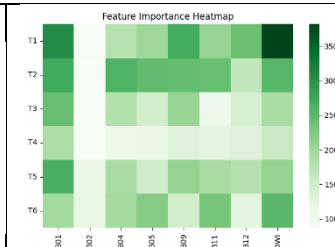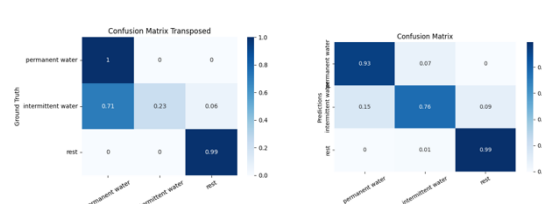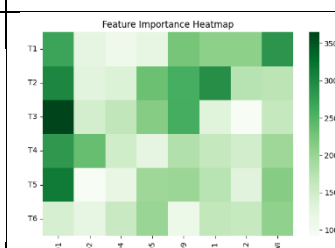
Table 2.6 shows the Classification Report, Confusion Matrix and Feature Importance scores for the Demonstration Areas specified in Table 2.1. Chile AOI had an extensive area, so subregion modelling was carried out by creating 3 separate models for the 3 sub-regions. Overall, the results show that Permanent Water class precision and recall was usually very high. For intermittent water, the model performs slightly deteriorated and is correlated with the number of samples available for training. When the number of samples for a class were higher than 6000, we saw very good results, otherwise, the model performed worse. The **rest** class is a class which combines the rest of the LC classes into a single class and the precision and recall for that class were near perfect because of the high amount of data available for the class.

The feature importance heatmap shows higher importance for B01, B05, B09 and NDWI S2 bands. However, the variance of the contribution was not very high as it can be seen that the lowest contributing band score was 100 and the highest contributing band score was between 350 and 400. This shows that the feature selection was done very carefully and the contribution of all the bands is used to make predictions.

Overall, the Water Pipeline is impressive at detecting water bodies wider and bigger than 10m. The model only appears to struggle only with very long and narrow water bodies with the width of the water body being around 10m.

Table 2.6 : Prediction statistics for Demonstration Areas at 10m resolution using S2 data
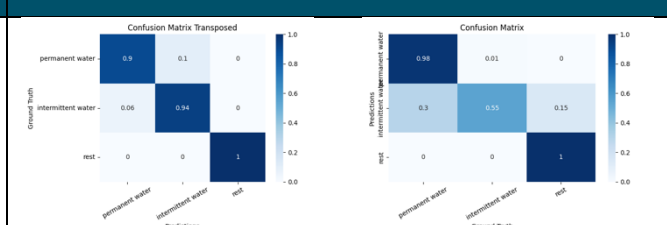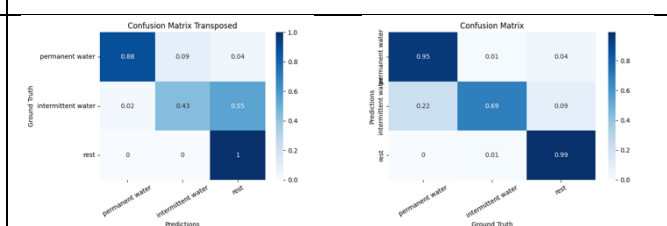
AOI	Classification Report	Confusion Matrix	Feature Importance
Chile-1 AOI	Chile-1 classification report (below)		
Chile-2 AOI	Chile-2 classification report (below)		
Chile-3 AOI	Chile-3 classification report (below)		
Albania AOI	Albania classification report (below)		
South Korea AOI	South Korea classification report (below)		

Chile-1 AOI
```
                    precision  recall  f1-score   support
permanent water       0.68      0.89     0.77     219051
intermittent water    0.61      0.21     0.32      72080
rest                  1.00      0.99     1.00   12820094

accuracy                                 0.99   13111225
macro avg             0.76      0.70     0.69   13111225
weighted avg          0.99      0.99     0.99   13111225
```

Chile-2 AOI
```
                    precision  recall  f1-score   support
permanent water       0.96      0.91     0.93     193133
intermittent water    0.12      0.81     0.21       4603
rest                  1.00      1.00     1.00    6226512

accuracy                                 0.99    6424248
macro avg             0.70      0.90     0.72    6424248
weighted avg          1.00      0.99     0.99    6424248
```

Chile-3 AOI
```
                    precision  recall  f1-score   support
permanent water       0.98      0.93     0.95     501828
intermittent water    0.10      0.61     0.17       5663
rest                  1.00      1.00     1.00    2469865

accuracy                                 0.98    2977356
macro avg             0.69      0.84     0.71    2977356
weighted avg          0.99      0.98     0.99    2977356
```

Albania AOI
```
                    precision  recall  f1-score   support
permanent water       1.00      0.91     0.95     423156
intermittent water    0.11      0.89     0.20       4726
rest                  1.00      1.00     1.00    1103626

accuracy                                 0.98    1531508
macro avg             0.70      0.93     0.72    1531508
weighted avg          0.99      0.98     0.98    1531508
```

South Korea AOI
```
                    precision  recall  f1-score   support
permanent water       0.99      0.95     0.97      53952
intermittent water    0.72      0.89     0.79       8271
rest                  1.00      1.00     1.00     183384

accuracy                                 0.98     245607
macro avg             0.90      0.95     0.92     245607
weighted avg          0.99      0.98     0.98     245607
```

Taiwan AOI		
Uruguary AOI		

2.2.5.2 Change Detection

Table 2.7 presents the classification report and confusion matrix for the region of Iraq and Syria for detection of changing water bodies. The details of the results are shared in the D5.4 Map Making use case Section 7. Two ML models were used, both trained over the S2 Data with bimonthly aggregated data, at 120m resolution and the second at 10m resolution. The model at 120m resolution was mainly utilized to drill down probable change. A second model trained on 2019 S2 data was applied to generate predictions for 2022 data. By utilizing Change Detection and Drill Down, only the necessary portions of the map are updated, reducing both operational and Sentinel-2 data costs. The quality of results is exceptionally high, especially for classes like water, demonstrating the effectiveness of change detection, as observed in the LC-CMS use case.

Table 2.7 : Prediction Statistics for Change Detection over Iraq and Syria AOI for detecting changing water

Input	Classification Report	Confusion Matrix
S2 Resolution -10m Year -2019		
S2 Resolution - 120m Year – 2019		

2.3 Built-Up

The following section provides a summary of the experiments performed for the Built-up use case.

2.3.1 Overview

This section discusses research done and results obtained within the Built-up area use-case. The research integrated existing knowledge with the GEM platform (eo-learn and eo-grow) to create a process for identifying new urban areas on a large scale.

Various methods were explored for built-up area detection, including:

- multi-class land cover classification,
- binary classification of built-up class,
- pixel fraction prediction through regression using pixel-based approach using LightGBM
- regression task with TFCN architecture
- classification task with TFCN architecture
- object-detection

All processes described here rely on eo-learn and eo-grow for reproducibility, configurability, and scalability. The latter was demonstrated for the entire continent of Africa due to the 120 m resolution's speed and cost-effectiveness. Results are accurate enough for change detection, and a drill-down mechanism identifies areas for higher-resolution processing to enhance accuracy in specific regions.

Within the use-case we have also developed and openly released a 120 m resolution global cloudless mosaic Analysis Ready Data cube for cost-effective research.

2.3.2 Areas of Interest

As part of demonstration phase, several AOIs have been selected, as shown in Table 2.8. The general development of eo-grow library, development of models and testing of various approaches has been done on data from Europe, most notably Slovenia, France and Azerbaijan, where we also have several options for ground-truth datasets, as described in detail in Section 3 of deliverable D5.2 – Built-up area use-case.

Table 2.8: AOIs used in the built-up use-case.

AOI	AOI Area (km2)	Activity Type	Resolution (m)	Prediction Year
Slovenia	20.273	Development	10, 120	2019, 2020
(Metropolitan) France	551,695	Development	10, 60, 120	2017, 2018, 2019, 2020

Azerbaijan	86,600	Development	10, 60, 120	2019
Africa	30,370,000	Demonstration	120, 10 (only parts)	2020

Figure 6: Built-up Area AOI used in demonstration phase, plotted using world cylindrical equal area projection to appreciate the size of area.

2.3.3 LightGBM binary classifier deployed on Sentinel Hub as evalscript

In Sections 4, 5 and 6 of D5.2, several tests have been performed to select the most suitable model type and parameters for performing cost-efficient large scale low-resolution built-up detection.

One of the goals of the exercise presented in section 4 of the D5.2 was to create a workflow where the processing Sentinel-2 pixels to discern built-up areas is pushed to the Sentinel-Hub. This way the classification can be done on demand, and at any time and place on Earth. In details, the approach is described in blog post **How to train a binary classifier for built-up areas**[2].

We allowed the feature selector to utilize all L2A Sentinel-2 bands as well as some interesting indices and feature ratios. In the end, a combination of raw bands (B01, B02, B03, B04) and several indices (NDVI, NDWI, NDVI_GREEN, NDVI_RE1, NDVI_RE2, NBSI, CL_GREEN, STI; a collection of well-established ones can be found in the public Sentinel Hub custom script repository[3]) has provided the best results.

The model is also deployed on Sentinel Hub, so the demonstration of this approach is available on EO-Browser at this link. Keep in mind, that the model was trained on Slovenia data, and while we have shown it works well on Slovenia data (shown in Figure 7), we expect it would struggle in dryer climates and in areas with different

---

[2] https://medium.com/sentinel-hub/area-monitoring-how-to-train-a-binary-classifier-for-built-up-areas-7f2d7114ed1c
[3] https://custom-scripts.sentinel-hub.com

soil types. It works on 10m resolution and performs reasonably well also when data at lower resolution is used, as can be seen in Figure 8.
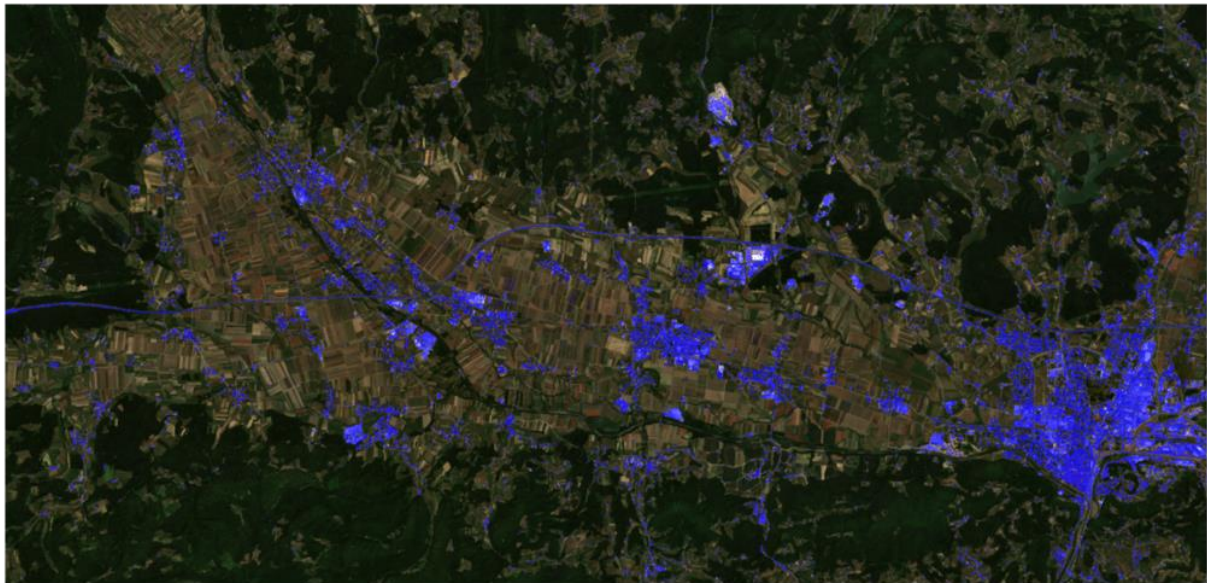
Figure 7: Built-up classifier mask script applied on Spodnja Savinjska Region, Slovenia, on 29.6.2921.
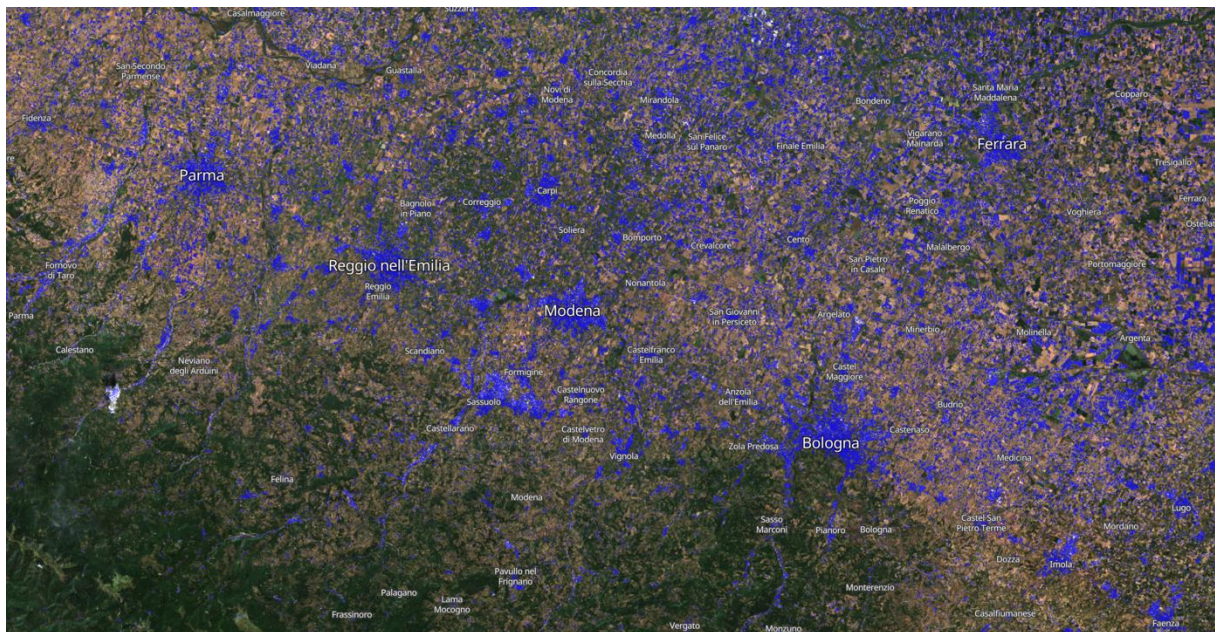
Figure 8: Built-up classifier mask script applied on low-resolution pixel values over Emiglia-Romagna region in Italy.

2.3.4 LULC classification

We have started exploring built-up area classification problem with a more general multi-class classification problem. Instead of using a simple binary-class classifier like shown in previous section, we have trained a LULC classifier predicting forest, grass, farmland, bare soil, sand, built-up, permanent and intermittent water and ice and snow land cover.

We have spent significant amount of time exploring how to prepare ground truth data (in details described in D5.2, but also in deliverables D2.1: Data management plan and D2.2: Ground Truth data collection process) to be suitable for 120m resolution, as at that scale, each pixel often a mixture of classes. The preparation of ground truth data, resampled to 120m, has also driven the experiments for best sampling methodology. We have finally decided on pixel-wise 5-fold cross-validation approach.

The results, obtained with a LightGBM classifier with default parameters, trained a small, hand-picked training data set using a yearly stack of all 12 bands from Sentinel-2 Global Mosaic is shown in Figure 9. Visually, the results are very good, indicating that a small number of highly accurate and balanced reference polygons could perform better than a large reference dataset of intermediate accuracy.
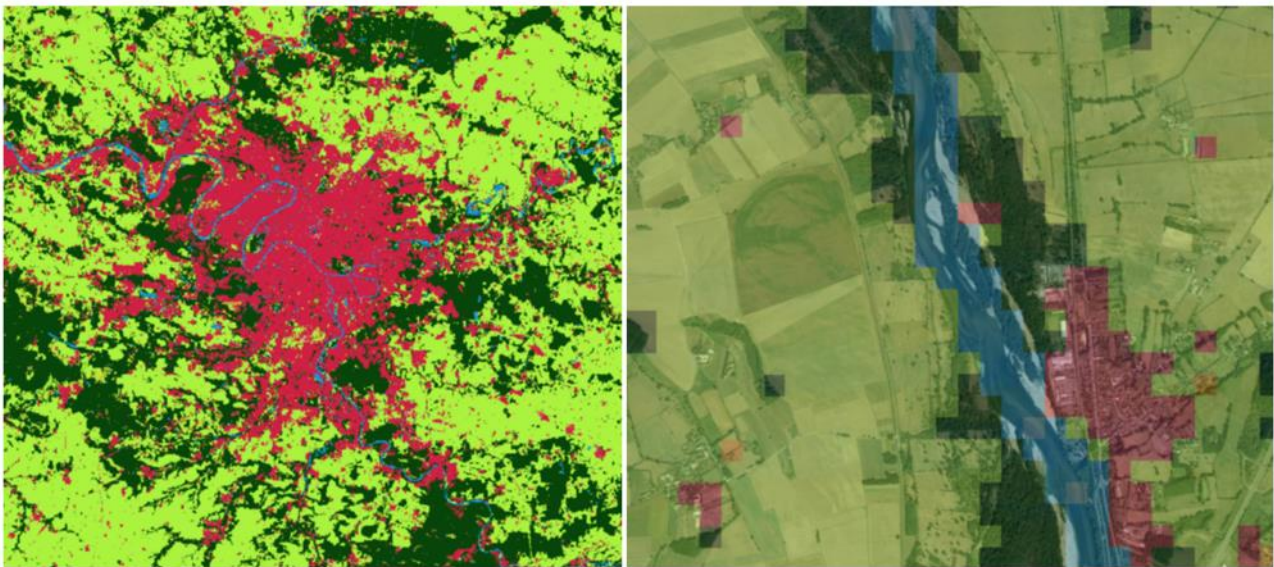
Figure 9: Results of random forest classification using TomTom ground truth in France.

To better understand how this model generalize we applied it on the entire Europe (Figure 10). Overall, the results were quite good in the Central Europe, where vegetation is similar to France.

While testing the effects of training data purity, we have found out that although the precision for mixed pixels improves at very high purity, the precision for all other classes drops. Since our task in the use-case is focused on built-up, the multi-class classification this seems to have this inherent issue, counterproductive to our use-case. Regardless, the experiments we have run on LULC classification have pinpointed to issues we could expect when running classification at 120 m resolution. The mixture of the LULC classes at this scale seems to be the limiting factor, hinting that predicting a fraction of each class in each pixel (e.g., regression) would probably the better approach.
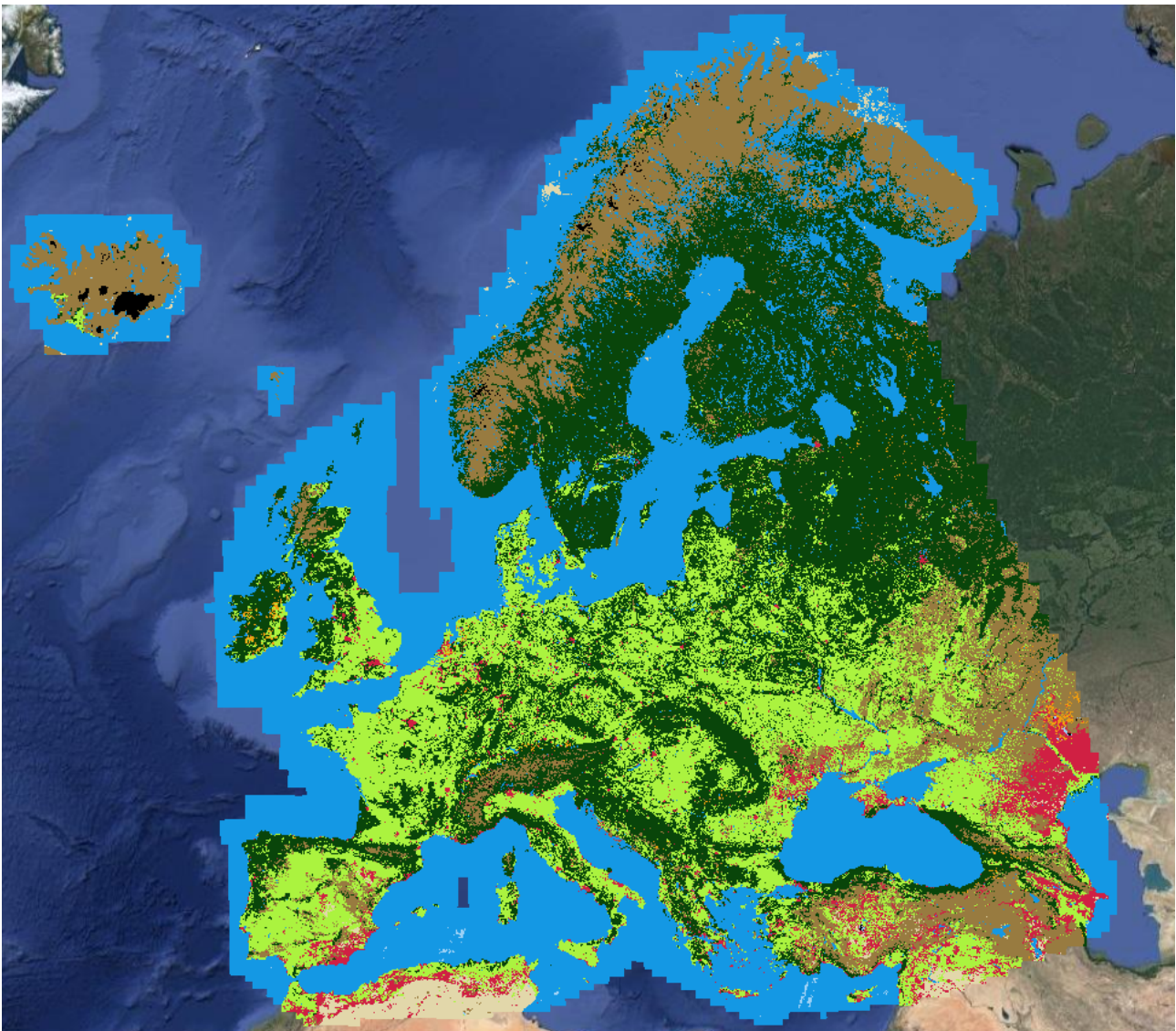
Figure 10: Model, trained in France, applied to whole Europe.

2.3.5 Built-up area regression models

With the regression approach we narrow down our objective to only distinguish between built-up and non-built-up areas. While a classification model would detect a presence of built-up, the aim of a regression model is also to approximate the fraction of built-up area in each pixel. Hence this way we can achieve a built-up detection on a sub-pixel level. Although we have focused exclusively on artificial surfaces, the approach could be used for other classes and even turned into multi-class fractional cover regression/classification task.

Based on our insights mentioned in previous section, we have tested two regression models. First one was pixel based LightGBM regressor, while the other was based on temporal fully connected network (TFCN) architecture.

2.3.5.1 LightGBM regression

In this approach we use the fractional land cover and try to predict what percentage of a pixel is inhabited by a given class. We have used the LightGBM regressor with default arguments and trained the model with respect to the RMSE (root mean squared error), MAE (mean absolute error) metrics and R^2 score for more insight.

Finally, we compared regression and classification results by using pseudo-probabilities of the classifications in the same way as regression predictions. Figure 11 compares:

- Pseudo-probabilities of binary classification model, trained to detect pixels with >30% built-up

- Pseudo-probabilities of multiclass classification model, trained on majority class ground truth

- Predictions of regression model without weights.
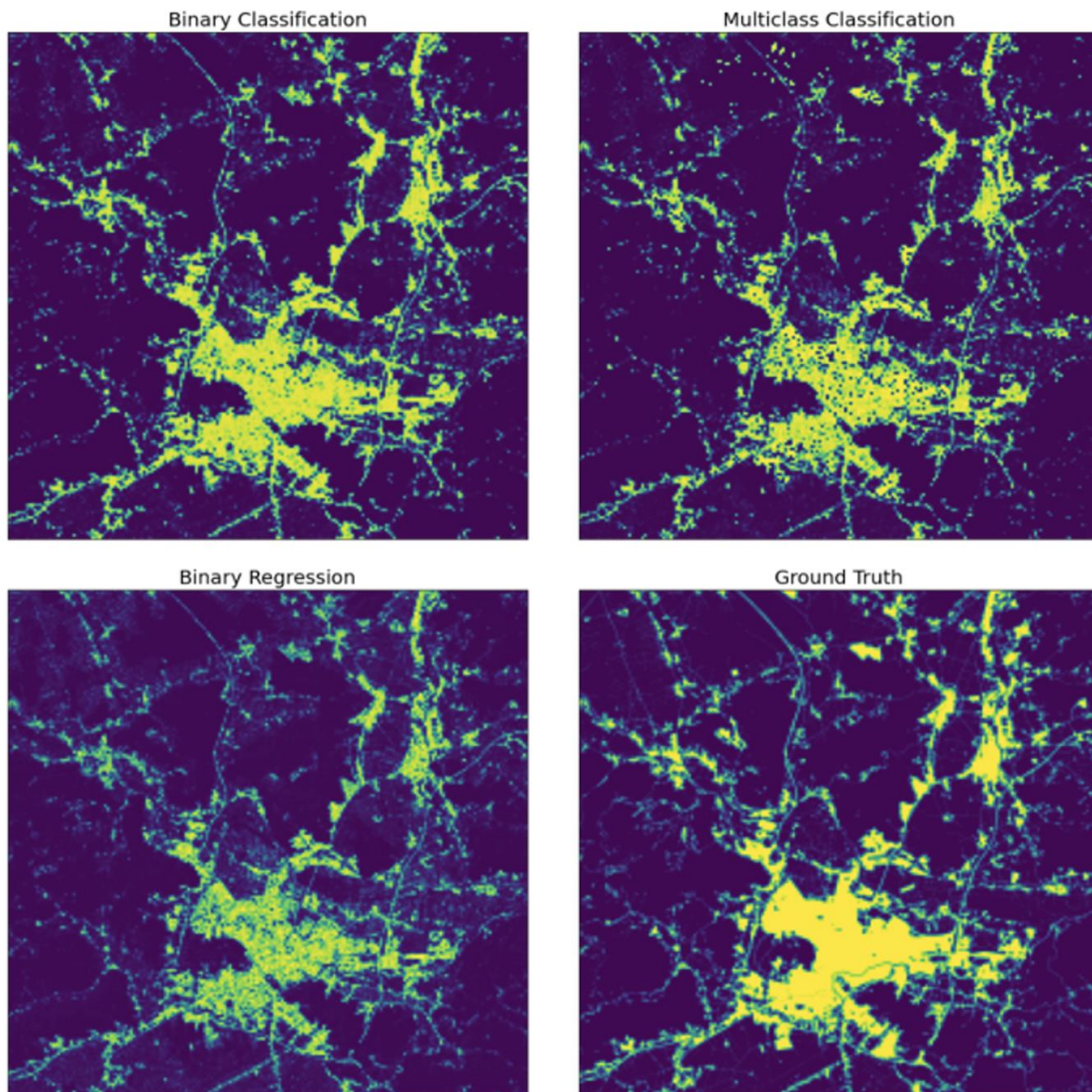
Figure 11: Comparison of various models for built-up areas.

The results show small differences, which should be evaluated from the perspective of down-stream application of the results. For instance, if we would like to use the results to find completely new areas being built, then the model should be highly confident in predicting pixels with low fraction of built-up areas. On the other hand, if we are looking for changes in already built-up areas (e.g., densification of buildings), then we probably do not care much for the low fraction pixels.

In Table 2.9 we compare results of LGBM regression model over France, trained on different ground truth data:

- Using Open Street Map (OSM) built-up reference, obtained by rasterizing OSM buildings and roads on 10m resolution and downsampling to 120m by converting to percentages.

- Global Human Settlements (GHS) data, sampled to 120m resolution by averaging percentages.

- A regression model trained on OSM built-up reference.

Overall, it seems that OSM reference data has a high accuracy in France. Some inaccuracies are caused by the process if mapping, buffering and rasterization. GHS is also quite accurate but, in a few areas, has issues with false positive detections. Our model has more false detections and in general we obtain more "averaged" predictions i.e., less low values and almost no high values.

Table 2.9: Comparison of several models, showcasing their drawbacks.

Model from OSM reference	Model from GHS data	Regression model from OSM
Regression model predicts significantly lower values in cities but non-zero values over the sea and other areas:		
Regression model predicts built-up in various cultivated and bare land areas:		
In some cases model from GHS data mistakes bare mountainous areas for buildings:		
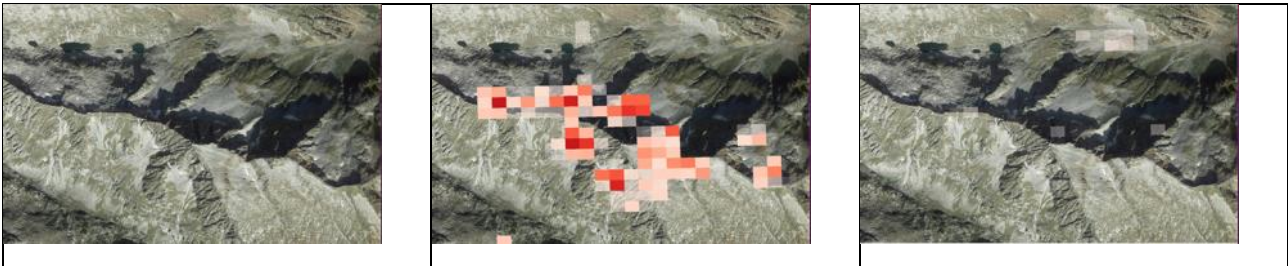
In this (rare in France) case OSM reference doesn't cover an airport:

In general we can see that the model from GHS data and regression model don't detect roads:

A military base which is not in OSM still gets predicted in the model from GHS data and regression model:

2.3.5.2 Temporal Fully Convolutional Network (TFCN) architecture

The experiments above reported were executed using a pixel-wise ML model, meaning that a built-up fraction value was estimated for each pixel separately and independently, depending only on the reflectance values of the pixel. While such approach has the advantage of being flexible and fast to train and run inference on, it fails to capture the contextual information provided by the neighbourhood of each analysed pixel. As the built-up areas exhibit specific contextual patterns, we evaluated the performance of the Temporal Fully

Convolutional Network (TFCN) architecture, proposed by our group for the joint spatial-temporal analysis of satellite imagery[4].

The distribution of the built-up fractional cover over Africa from rasterized Google buildings dataset used as ground truth data is heavily skewed towards zero (not built-up) and this can lead to a very low number of samples containing larger built-up percentages that are required for the model to learn how built-up areas look like. After the sampling, the train-test-validation split has been performed and can be seen in Figure 12.
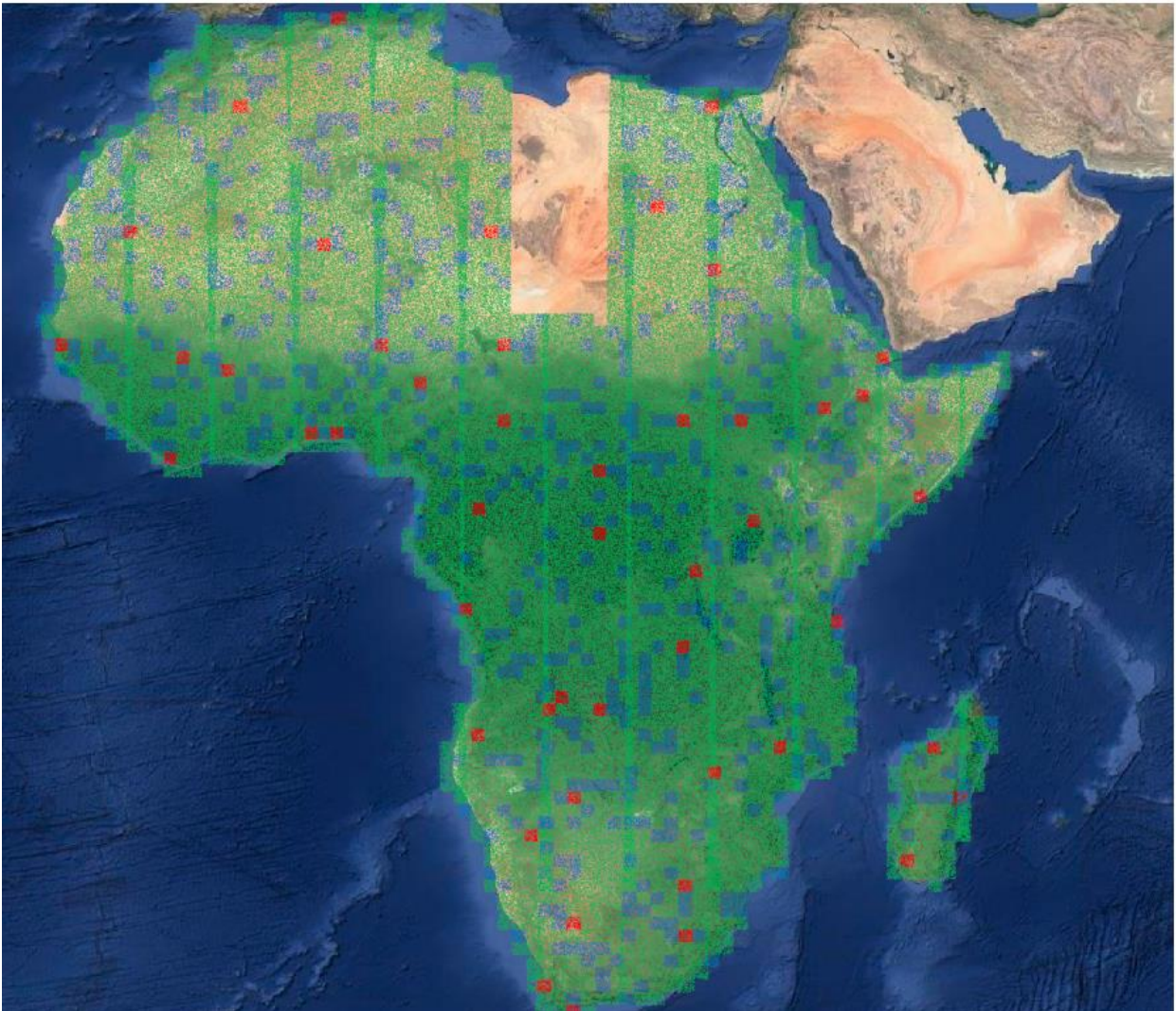
Figure 12: Train (green), cross validation samples (blue) and the test set samples (red).

The FCN architecture, based on U-net[5], is de facto state-of-the-art architecture in semantic segmentation of natural and remotely sensed images. While the backbone of the FCN consisting of an encoding and decoding part are maintained, the convolutions in the encoding path of the TFCN are 3D rather than 2D. This modification allows the network to learn spatial-temporal features simultaneously, therefore characterising

---

[4] Lubej M. et al. "Spatio-Temporal Deep Learning: An Application to Land Cover Classification", LPS 2019, 10.13140/RG.2.2.17961.65128

[5] Ronnenberg O., et al. "U-Net: Convolutional Networks for Biomedical Image Segmentation", CVPS 2015, 10.48550/ARXIV.1505.04597

built-up areas both in the spatial and temporal dimension. Code for the TFCN architecture is open-source and can be found in sentinel-hub/eo-flow[6] repository on GitHub.

We experimented with different loss functions, to include both regression and classification task simultaneously during optimisation of the model. This was done in two different ways:

- by using a single output and two losses, a Mean Squared Error (MSE) loss for the regression task and a Binary Cross Entropy (BCE) loss for the classification task. The two losses are then combined in a weighted sum to ensure they equally contribute to the total loss.
- by using two separate outputs and applying a MSE loss on the output of regression task and a BCE loss on the classification task. The difference in this case is that the two separate outputs are allowed to vary independently, focusing on the different tasks.

The dataset is highly imbalanced, and we had to under-sample the number of patchlets with very low (or zero) amount of built-up data.

2.3.6 Built-up Object-detection

For demonstrator purposes we have run a model we created as part of the ESA Philab QueryPlanet 4000124792/18/I-BG grant.

The model relies on HIECTOR (HIErarchical object deteCTOR)[7], a tool to apply object detection on satellite imagery of varying spatial resolutions in a hierarchical fashion. The prototype developed uses Sentinel-2, Airbus SPOT and Airbus Pleiades as data sources for a hierarchical detection at three spatial scales. While the tool can be used for the detection of different man-made objects, we validated HIECTOR on building detection.

To facilitate the generalisation and application of the task at different scales we formulate the task as the detection of oriented objects, with the direct estimation of oriented bounding boxes (OBB). We built on the Single-Stage Rotation-Decoupled Detector for Oriented Object (SSRDD) architecture and used a pretrained ResNet34 backbone to train a model on Sentinel-2 imagery and a joint model on Pleiades/SPOT imagery. The models take as input 4-channels images, namely the blue, green, red, and infrared bands.

Within GEM, we have taken the open-sourced model and used eo-grow to scale up running the model on two countries from Sahel region: Eritrea and South Sudan. An example from South Sudan is shown in Figure 13. The red squares are oriented bounding boxes over detected (groups of) buildings.

---

[6] https://github.com/sentinel-hub/eo-flow/blob/70a426fe3ab07d8ec096e06af4db7e445af1e740/eoflow/models/segmentation_unets.py#L129C13-L129C13
[7] https://github.com/sentinel-hub/hiector
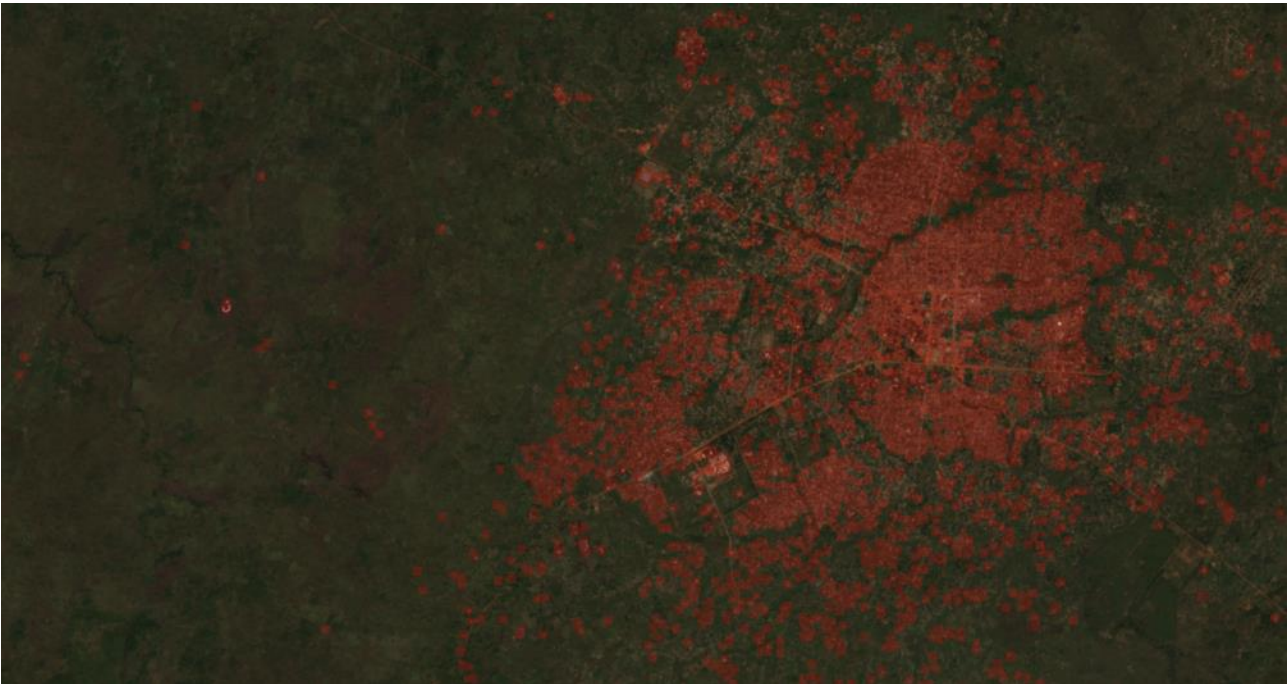
Figure 13: Oriented bounding boxes over Yei, a medium-sized city in South Sudan's southwest.

2.3.7 Demo over Africa - comparison of results

For demonstration purposes we have re-trained or fine-tuned the models on Africa. We continued to use the 120 m Sentinel-2 global mosaic (12 bands at 36 time points) for 2020 to show the approach on a cost-effective scale. For reference data we rasterized Google buildings with a 5m buffer at 10m resolution and then transformed those results to a built-up fraction mask at 120m. The buffer was again necessary to achieve a better 120m mask. Some areas of Africa are not covered by the Google buildings reference, which we had to account for. The lessons learned while developing the model have been collected in another blog post: **Challenges of large open-source datasets for building detection in Africa[8].**

In Table 2.10 we compare the results of the two best models with regression: LGBM and TFCN. Overall, both models produce good results, with LGBM typically predicting lower fractions and more salt and pepper results. The built-up landcover class is as such one of the most difficult ones to predict at 120 m resolution, and even more so in Africa, where the spectral difference of built up is often not different from bare soil.

Table 2.10: Comparison of several models, showcasing their drawbacks.

Sentinel-2 120 m mosaic	LightGBM regression	TFCN regression
Bechar, Algeria: False detections in the mountainous and arid areas can be seen in both models, with the TFCN model slightly more robust to these false detections		

---

[8] https://medium.com/sentinel-hub/challenges-of-large-open-source-datasets-for-building-detection-in-africa-4d1376f85894

Bossangoa, Central African Republic: The TFCN regression model predicts a more uniform city shape compared to the LGBM regression. It also displays less less salt and pepper effects, likely due to the convolutions.

Kahuzi Biega Park, DRC: Areas covered with forest do not seem to cause many false positive detections. Buildings can be missed if they are covered by tree canopy.

Makongo, Ghana: LGBM predictions are more scattered. Due to the spatial resolution of the imagery (even on 10m) it's hard to visually assess the the corectness of the predictions in sparsely populated areas.

Windhoek, Namibia: Larger cities are consistently found up by all the models.
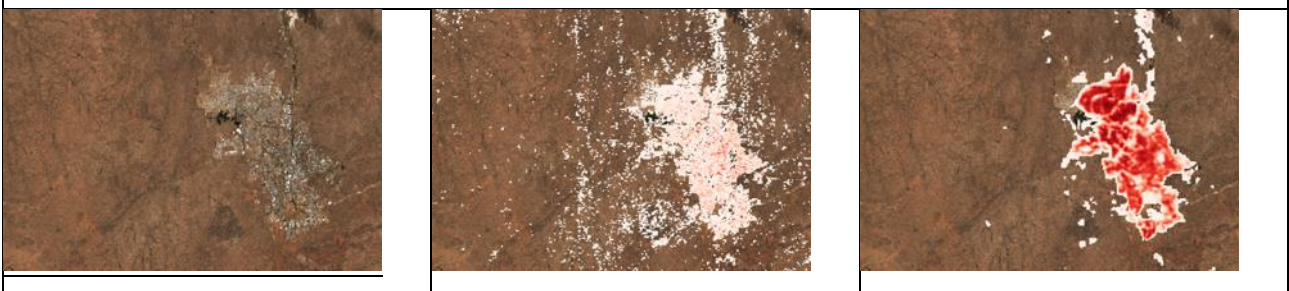
Medina Gounass, Senegal: Even when the difference between built-up and surrounding (bare soil?) areas are difficult to discern, the models seem to pick it up, particularly the TFCN.
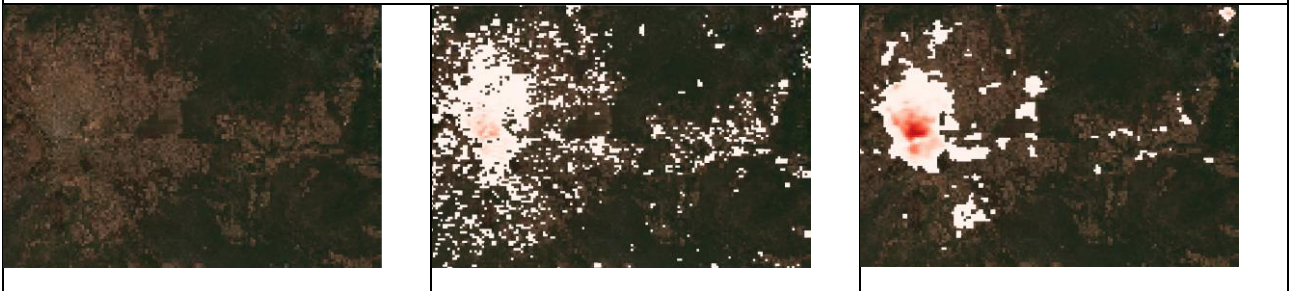
2.4 Crop Identification Demonstration

2.4.1 Overview

When project results could not significantly increase the accuracy of crop identification with weather data (D5.6), the consortium de-emphasised the crop identification demonstrator. Instead, we focused on hyper-resolution city heat maps as a demonstrator for the downscaling of temperature fields in built-up areas with very high resolution satellite data and local measurements (D2.3, Section 2.4; D2.4, Section 3.6). In response to requests from project partners we added support for the computing of climate variables on grids that can be consumed by EO machine learning applications (D2.4, Section 3.3). For meteoblue internal use we created an interface for working with measurement data (D2.4, Section 2.2).

Some text in these sections provides publicly available information extracted from project confidential deliverables.

2.4.2 Crop Identification use case

The crop identification use case was intended demonstrate improved accuracy when using weather data (Figure 14) in combination with EO data (Figure 15). As this could not be achieved with the available ground truth data for crops in Europe, the demonstrator development was restricted to allowing users to inspect the results of a classification run on an interactive map. The example chosen for the demonstrator covers Slovenia, for which the best ground truth data was available.
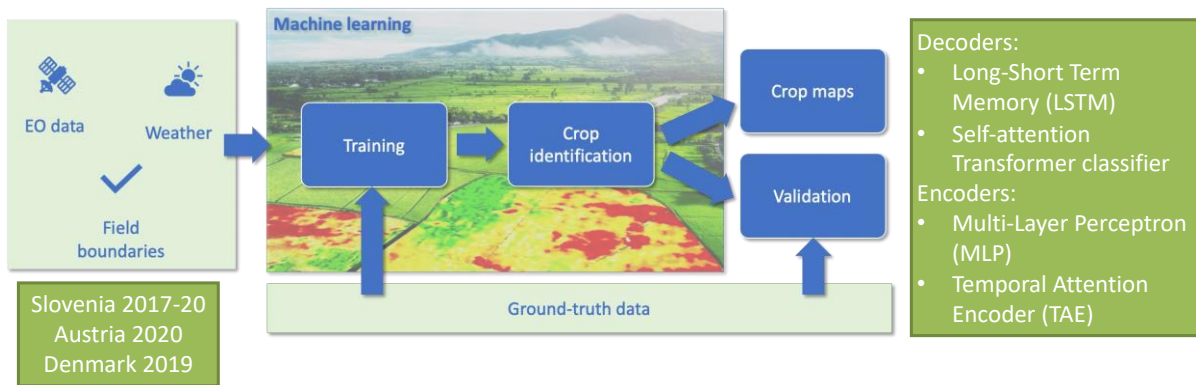
Figure 14: The crop identification use case. Note weather variables are used as parameters.
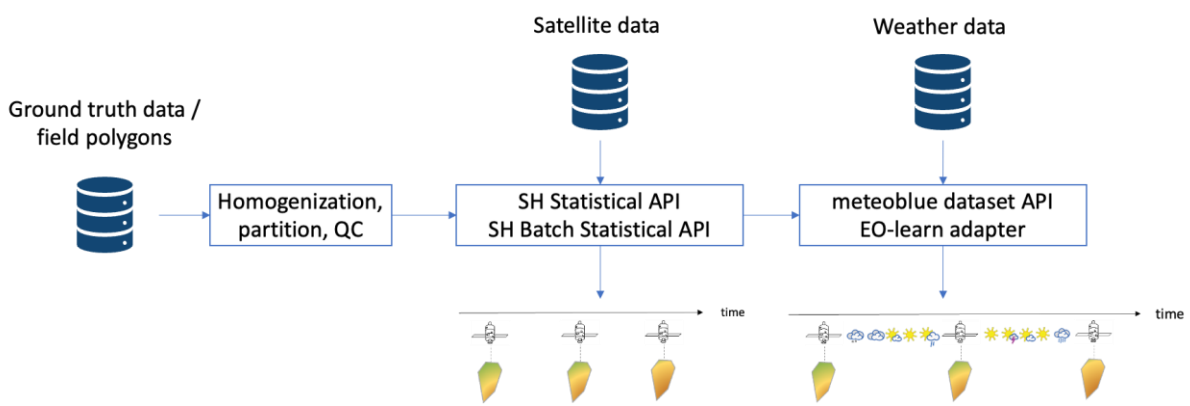
Figure 15: Data preparation workflow taking advantage of EO-learn.

The data preparation process heavily relies on Sentinel Hub's Batch Statistical API and meteoblue Dataset API. A flexible encoding-decoding framework (FlexMod) was designed to facilitate experiments with different models, features, and configurations.

To handle the multi-dimensional layered data described in D5.6, Section 4, we use a flexible encoding-decoding framework (namely FlexMod, designed with TUM during the project): multiple encoders are designed for features of different shape such as time length, discretization, or dimension (namely for the EO data and weather variables) and are then passed to the decoder via a mediator, as shown in Figure 16.
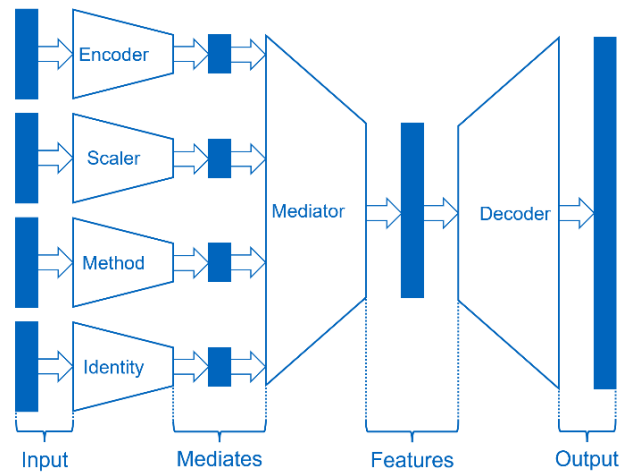
Figure 16: Basic concept of the FlexMod Framework fostering flexibility and standardization. Blue bars denote data or tensors.

To perform crop identification, we tried different encoder-decoder configurations: a Long-Short Term Memory implementation, a Transformer Classifier, Multi-layer Perceptron and Temporal Attention encoders. The best performing model was the Transformer Classifier decoder with Temporal Attention feature encoders (D5.6, Section 10). A result is shown in Figure 17.

Figure 17: Using a Sentinel-2 true colour image retrieved from Sentinel Hub services as background.

2.4.3 Temperature downscaling use case

The polar-orbiting satellite Sentinel-2 is used to calculate normalized difference vegetation index (NDVI), normalized difference built-up index (NDBI) and albedo at a horizontal resolution of 10 m (Figure 18). Therefore, different bands of the satellite are used:

$$NDVI = \frac{Band\ 8 - Band\ 4}{Band\ 8 + Band\ 4}$$

$$NDBI = \frac{Band\ 11 - Band\ 8}{Band\ 11 + Band\ 8}$$

$$Albedo = 0.356b_2 + 0.13b_4 + 0.373b_8 + 0.085b_{11} + 0.072b_{12} - 0.0018$$

where *b2, b4, b8, b11* and *b12* are empirical coefficients.

The polar-orbiting satellite Landsat-8 is used to calculate land surface temperature (LST):

$$LST = \frac{K_2}{\ln(\frac{K_1}{TOA} + 1)}$$

where *K1* and *K2* are empirical coefficients, and *TOA* is the top of atmosphere signal.

The satellite information was obtained during clear sky summer conditions for more than 70 different city domains.
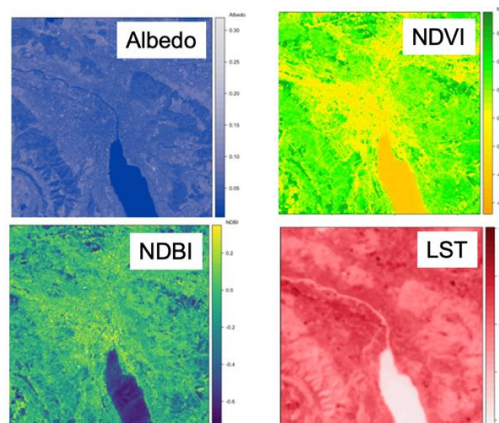
Figure 18: Albedo values (upper left), NDVI (upper right), NDBI (lower left) and LST (lower right) based on a satellite dataset for Zurich from 2019-07-29.

Figure 19 shows the high-resolution air temperature field in Zurich for different days based on 4 different satellite datasets.

The analysis showed that the sensitivity to different satellite datasets is small with a mean error below 0.04 K for all 4 datasets. Therefore, a static satellite dataset is used for the model predictors, instead of a dynamic satellite dataset.
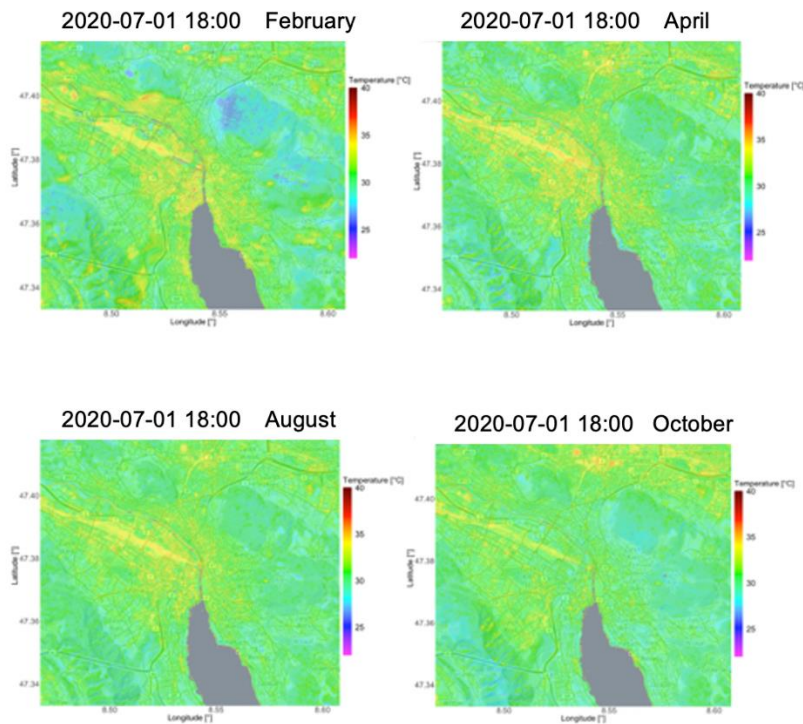
Figure 19: Air Temperature [°C] for Zurich for 2020-07-01 18:00 by using satellite data from February (upper left), April (upper right), August (lower left) and October (lower right).

These calculations are easily performed using eo-learn and eo-grow. As one application, urban heat islands (UHI) can be mapped out by

1. calculating the difference between each pixel and the mean values of the surrounding rural areas, which are defined by the classification of the local climate zones (LCZs);

2. averaging the differences over one year.
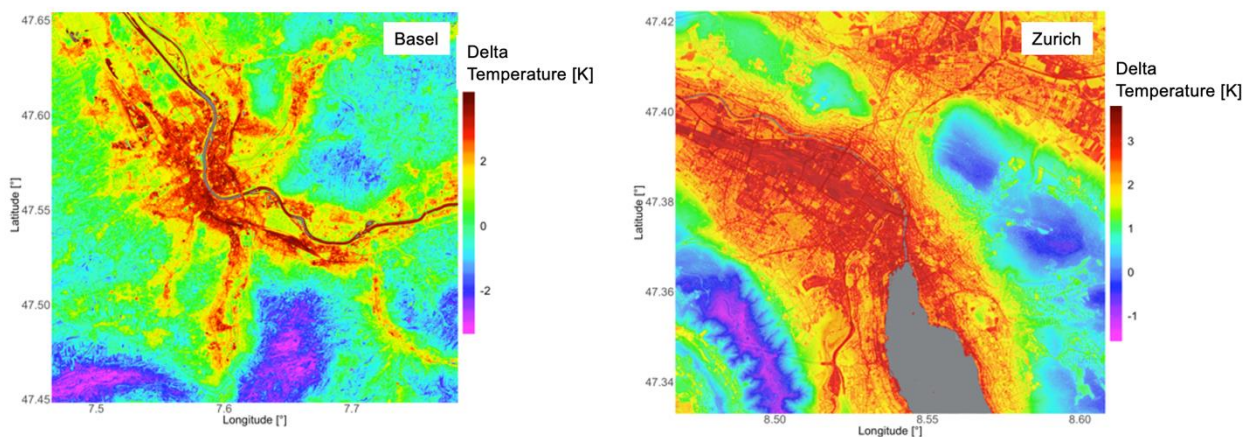
Figure 20: UHI for Basel (left) and Zurich (right) by averaging air temperatures over the entire year 2020.

The approach enables Continuous Monitoring using just the weather data. Downscaling model updates are required from time to time to adjust to changes in the built-up environment that change the indices used in the model.

2.4.4 Compute climate variables on grids use case

In discussions with project partner the need for derived weather and climate variables became apparent. Derived here means that several "raw" variables need to be combined to obtain a desired variable. meteoblue supports this through a calculation engine in its Dataset API and, by extension, through its Python SDK and eo-learn adapter (D2.4, Section 3.4). The calculations of derived variables benefit from the optimisations described in D2.4, Section 3.5, making them more practical for larger datasets.

The following sections describe the derived variables and their calculation. The first example provides all technical details, which are the same for all derived variables save a few parameters. Full examples for all derived variables are given in Jupyter Notebooks[9].

2.4.4.1 Climate Indices

"A climate index is defined as a calculated value that can be used to describe the state and the changes in the climate system" – [Integrated Climate Data Center (ICDC)](#).

2.4.4.1.1 Tropical Nights

A tropical night is a term used to describe days when the temperature does not fall under 20 °C during the night-time. Due to climate change, many countries are experiencing a significant increase in tropical nights compared to last century.

Query for yearly number of tropical nights for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Temperature, 2 m above ground, hourly.

3. Aggregate by day to daily **minimum**.

4. Value is greater than or equal to **20** then pick **0 or 1**.

5. Aggregate by year to yearly **summation**.

Note: if producing a map or if the ERA5T grid-cell is in the immediate vicinity of the sea remember to mark the option "exclude sea points".

---

[9] https://github.com/sentinel-hub/eo-learn-examples/tree/main/climatological_days
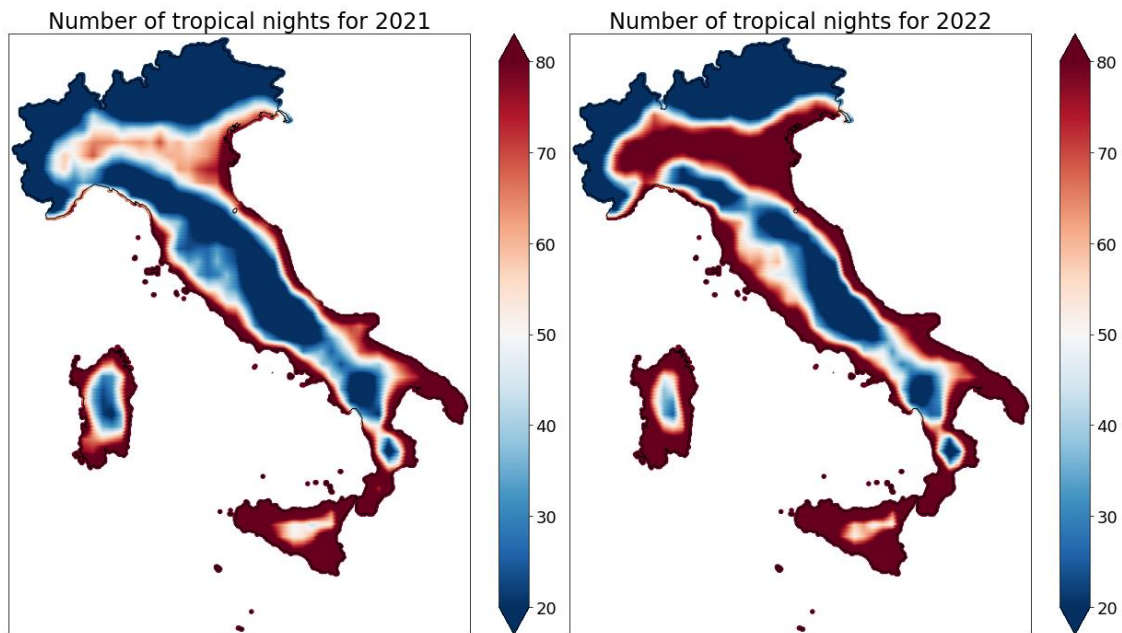
Figure 21: Number of tropical nights in Italy in two different years

The images in Figure 21 were obtained through Python, simply by selecting from the Dataset API a geographic polygon (in this case Italy), instead of a single location. The following option included in the available transformations was also used to obtain a better representation:

"Resample to a regular grid using linear interpolation", with a grid resolution of 0.025°.

Using the **meteoblue_dataset_sdk** Python package it is possible to retrieve data for creating the above maps in a few simple steps, defining a unique query. The tropical nights' example is shown in the following. All other examples are formed analogously.

Query for tropical nights:

```
query_map = {
    "units": {
        "temperature": "C",
        "velocity": "km/h",
        "length": "metric",
        "energy": "watts"
    },
    "geometry": {
        "type": "GeonamePolygon",
```

```
    "geonameid": 3175395 #this code corresponds to Italy
},
"format": "JSON",
"timeIntervals": [
    str(year)+"-01-01T+00:00/"+str(year)+"-12-31T+00:00"
],
"timeIntervalsAlignment": "none",
"queries": [
    {
        "domain": "ERA5T",
        "gapFillDomain": None,
        "timeResolution": "hourly",
        "codes": [
            {
                "code": 193,
                "level": "2 m above gnd"
            }
        ],
        "transformations": [
            {
                "type": "aggregateDaily",
                "aggregation": "min"
            },
            {
                "type": "valueIsAbove",
                "valueMin": 20,
                "returnClassification": "zeroOrOne"
            },
            {
                "type": "aggregateYearly",
```

```
        "aggregation": "sum"
      },
      {
        "type": "spatialTransform",
        "gridResolution": resolution,
        "interpolationMethod": "linear",
        "spatialAggregation": "mean",
        "disjointArea": "discard",
        "elevationDownscale": "disabled"
      }
    ]
  }
  ]
}
```

API call:

```
client = meteoblue_dataset_sdk.Client(apikey)

result = client.query_sync(query_map)

lats = list(result.geometries[0].lats)

lons = list(result.geometries[0].lons)

#Use a function to convert the datasetAPI protobuf file into a pandas dataframe

data = list(meteoblue_result_to_dataframe(result.geometries[0])["193_2 m above gnd_sum"])
```

2.4.4.1.2 Frost Days

Frost days are defined as days in which temperature drops below 0°C.

Query for yearly number of frost days for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Temperature, 2 m above ground, hourly.

3. Aggregate by day to daily **maximum**.

4. Value is less than or equal to **0** then pick **0 or 1**.

5. Aggregate by year to yearly **summation**.

Note: if producing a map or if the ERA5T grid-cell is in the immediate vicinity of the sea remember to mark the option "exclude sea points".

2.4.4.1.3 Ice Days

Ice days are defined as days in which temperature never goes above 0°C.

Query for yearly number of ice days for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Temperature, 2 m above ground, hourly.

3. Aggregate by day to daily **maximum**.

4. Value is less than or equal to **0** then pick **0 or 1**.

5. Aggregate by year to yearly **summation**.

Note: if producing a map or if the ERA5T grid-cell is in the immediate vicinity of the sea remember to mark the option "exclude sea points".

2.4.4.1.4 Hot Days

Hot days are defined as days in which temperature rise above 30°C.

Query for yearly number of hot days for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Temperature, 2 m above ground, hourly.

3. Aggregate by day to daily **maximum**.

4. Value is greater than or equal to **30** then pick **0 or 1**.

5. Aggregate by year to yearly **summation**.

Note: if producing a map or if the ERA5T grid-cell is in the immediate vicinity of the sea remember to mark the option "exclude sea points".

- Heating Degree Days (HDD) & Cooling Degree Days (CDD)

Heating degree days are a measure of how much (in degrees), and for how long (in days), the outside air temperature was below a certain level. They are commonly used in calculations relating to the energy consumption required to heat buildings. With regard to heating degree days, the *base temperature* of a building is the outside air temperature below which that building needs heating.

Cooling degree days are a measure of how much (in degrees), and for how long (in days), the outside air temperature was *above* a certain level. They are commonly used in calculations relating to the energy consumption required to *cool* buildings.

Further explanations can be found at https://www.degreedays.net/introduction.

Hot days are defined as days in which temperature rise above 30°C.

Query for yearly number of heating degree days for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Temperature, 2 m above ground, hourly.

3. Value is less than or equal to **15** then pick **0 or delta**. (Here 15 is the base temperature, which can be arbitrarily selected by the user).

4. Aggregate by year to yearly **summation**.

5. Download the file as CSV and divide the values by 24.

Note: if producing a map or if the ERA5T grid-cell is in the immediate vicinity of the sea remember to mark the option "exclude sea points".

2.4.4.1.5 Consecutive Dry Days

Consecutive dry days are defined as the number of consecutive days with less than 1mm of rain up to a maximum of 365 (1 year). This index is really important for evaluating drought conditions.

Query for maximum number consecutive dry days in a year for a given location:

1. Select a location and a time interval of 1 or more years.

2. Select the dataset ERA5T, Precipitation, Surface, daily sum.

3. Value is less than or equal to **1** then pick **0 or consecutive count**.

4. Aggregate by year to yearly **maximum**.
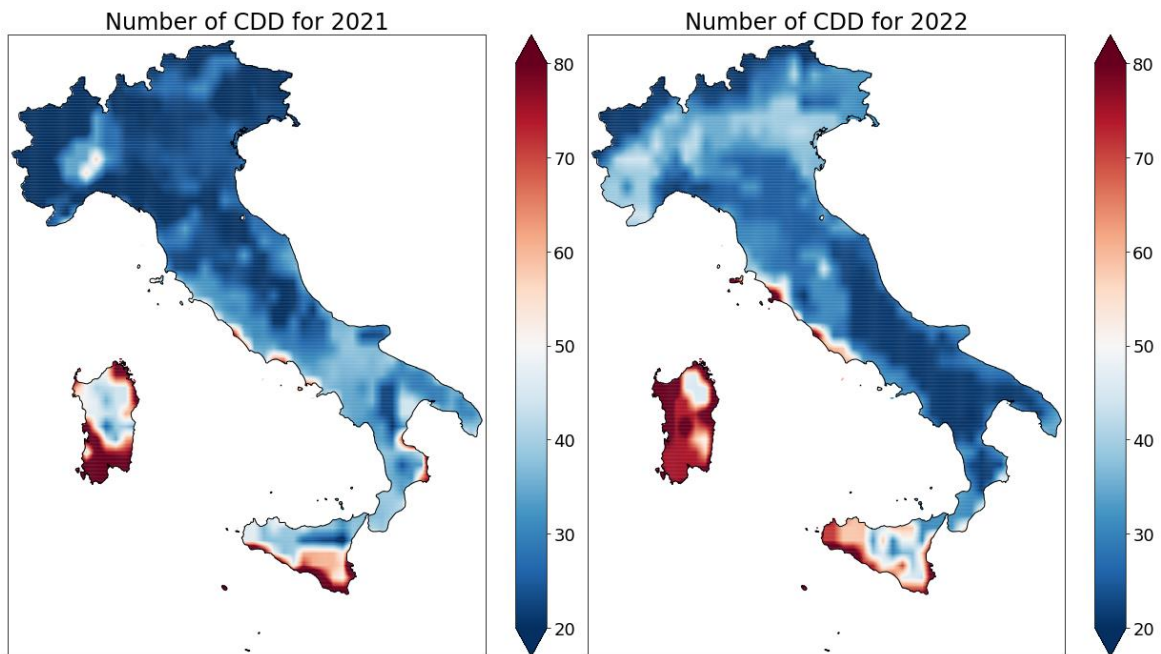
Figure 22:Number of cooling degree days (CDD) in Italy in two different years

The graphs above were obtained through Python, simply by selecting from the Dataset API a geographic polygon (in this case Italy), instead of a single location. The following option included in the available transformations was also used to obtain a better representation:

"Resample to a regular grid using linear interpolation", with a grid resolution of 0.025°.

2.4.4.1.6 Heavy Precipitation Days

Counting the number of days within a given period in which daily cumulative precipitation has exceeded a given threshold (such as 20, 30, 40 mm).

Query for number of heavy precipitation days in a year for a given location:

1. Select a location and a time interval of 1 year.

2. Select the dataset ERA5T, Precipitation, Surface, daily sum.

3. Value is greater than or equal to **20/30/40** then pick **0 or 1**.

4. Aggregate by year to yearly **summation**.

- Violent Precipitation Events

Counting the number of events in one year during which the cumulative precipitation in 3 days exceeded 150 mm. This index is very important because it provides an estimate of the meteorological contribution to the hydrological risk of a given area.

Query for number of violent precipitation events in a year for a given location:

1. Select a location and a time interval of 1 year.

2. Select the dataset ERA5T, Precipitation, Surface, daily sum.

3. Aggregate over a sliding time window using a running **Summation** of the last **3** timesteps.

4. Value is greater than or equal to **150** then pick **0 or 1**.

5. Aggregate by year to yearly **Summation**.

2.4.4.2 Other derived variables

2.4.4.2.1 Atmospheric Stability

Compute the heavy storm likelihood, i.e., the number of days in which the CAPE (Convective Available Potential Energy) value is above a certain threshold (e.g., 5000 J/kg).

1. Select a location and a time interval of 1 year.

2. Select the dataset ERA5T, CAPE, 180-0 mb, daily max.

3. Value is greater than or equal to **5000** then pick **0 or 1**.

4. Aggregate by year to yearly **summation**.

2.4.4.2.2 Agricultural Photovoltaic (Agro PV)

Agro PV is the simultaneous use of areas of land for both solar panels and agriculture. Because solar panels and crops must share the sunlight, the design of Agro PV facilities may require trading off such objectives as optimizing crop yield, crop quality, and energy production. In some cases, crop yield increases due to the shade of the solar panels mitigating some of the stress on plants caused by high temperatures and UV damage.

Thanks to the wealth of datasets provided by our API, we sought to construct a parameter expressing the suitability of a country to host Agro PV plants. The project is still under development, but the following steps give an idea of the potential of this approach.

From the GLOBCOVER dataset [0.3 km], we selected only the land cover types which may be suitable for installing PV panels. The list is:

11	Post-flooding or irrigated croplands (or aquatic)
14	Rainfed croplands
20	Mosaic cropland (50-70%) / vegetation (grassland/shrubland/forest) (20-50%)
30	Mosaic vegetation (grassland/shrubland/forest) (50-70%) / cropland (20-50%)
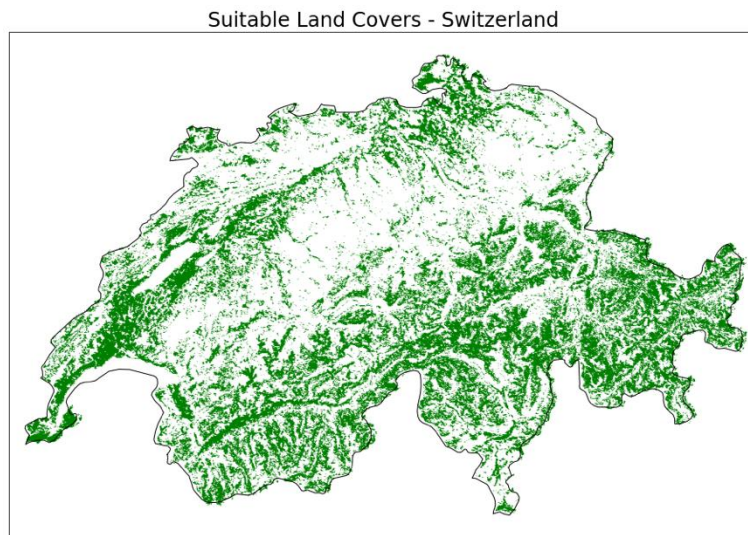
Figure 23: Ground cover in Switzerland that may be suitable for installation of AgroPV

From the NEMS4 model [4 km], we selected soil moisture 0-10 cm, averaged over a period of 10 years (2011 – 2020).
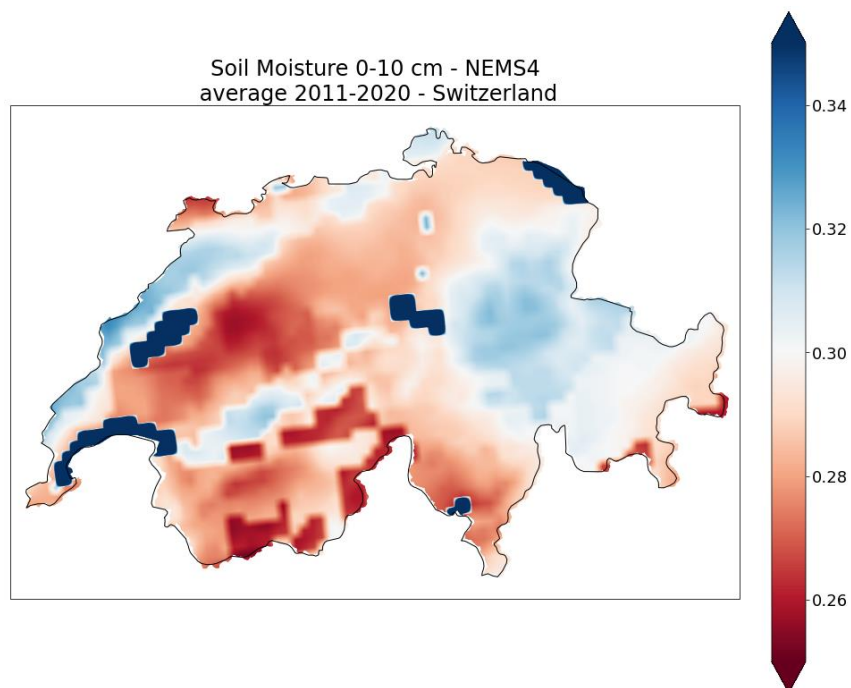
Figure 24: Soil moisture average Switzerland for the period of 2011 to 2020

We tried to come up with a parameter for Agri PV suitability combining the previous two data frames and adding some conditions:

- If soil moisture is > 0.4 then it's not suitable: too wet, need of intense solar radiation.

- If soil moisture is < 0.2 then it's not suitable: too dry.

In the range between 0.2 and 0.4 the more suitable terrains are the drier ones (more potential improvement from agro PV, therefore define the suitability as proportional to (1 – soil moisture).
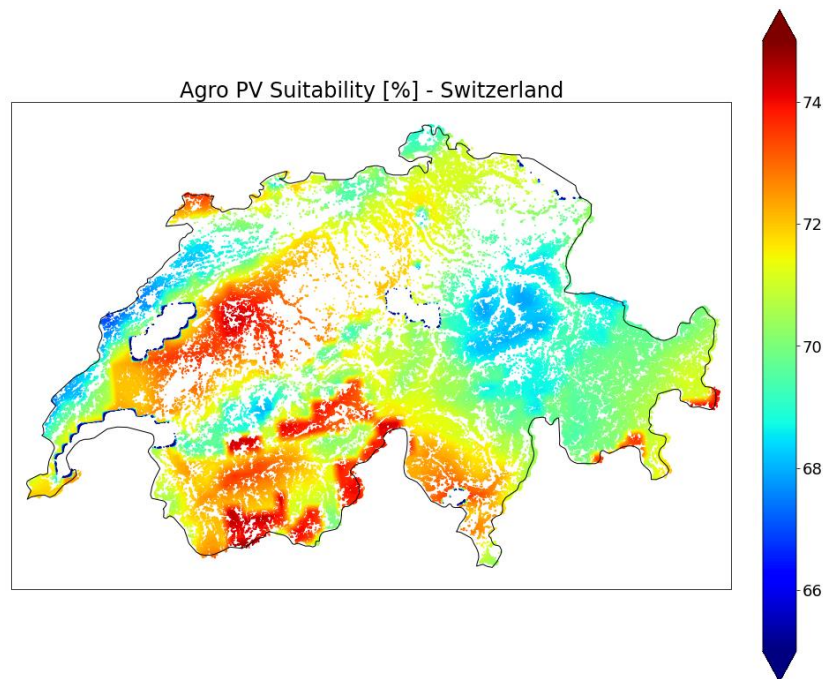
Figure 25: Areas in Switzerland with varying degrees of suitability for the installation of AgroPV

2.4.4.2.3 Difference of elevation in ERA5T grid-cells

This use-case shows how a static dataset (in this case height/elevation at 250 m resolution) can be used to obtain information about other datasets such as ERA5T.

The goal here is to obtain the maximum difference in heights within a given ERA5T cell:

1. Select a given location or area.

2. Select the dataset GMTED250, height/elevation, surface, static.

3. Resample to domain ERA5T using linear interpolation to downscale and Minimum to upscale values.

4. In parallel, select again the same dataset and resample to domain ERA5T using linear interpolation to downscale and Maximum to upscale values.

5. Subtract the two values to get the difference in elevation.
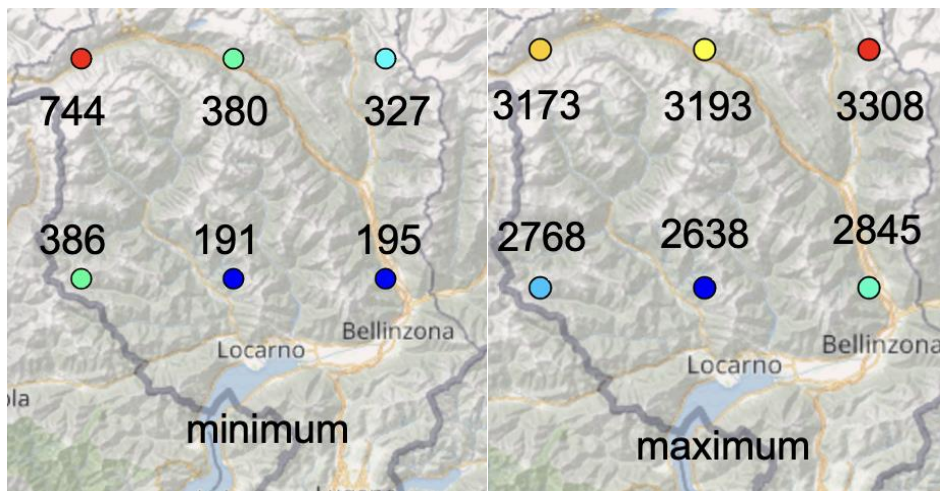
Figure 26: Minimum and maximum elevations (in meters) of ERA5T grid cells in Ticino

2.4.4.2.4 Land-use of ERA5T grid cells

Using the static dataset GLOBCOVER (resolution 300 m), it is possible to select a given land cover type and see which is the percentage of it within an ERA5T grid-cell.

1. Select a given area.

2. Select the dataset GLOBCOVER, Land Cover Classification, Surface, Static.

3. Value is between **xxx** and **xxx** then pick **0 or 1**. (xxx is a general land cover type, such as 190, 11 or 20 – can find the list here: https://docs.meteoblue.com/en/meteo/data-sources/datasets#globcover)

4. Resample to domain ERA5T using Linear interpolation to downscale and mean to upscale.

2.4.4.3 Measurements interface use case

Motivated by GEM needs for reliable ground truth data, meteoblue has created a web interface for measurement data for its own data management and as basis for future layered services such as Quality Assurance as a Service.

2.4.5 Demonstration Applications

The city heat map demonstrator has been developed into a product and is publicly accessible; other demonstrators require an API key[10] or, in one case, a specific arrangement for data protection. Please see Section 4 for details.

---

[10] The APIKEY for reviewers is '5d2e1945322c-GEM-reviewers'. For questions please contact support@meteoblue.com.

2.5 Conflict Pre-Warning Demonstration

The section below provides a summary of the experiments performed for the Conflict Pre-Warning demonstration on the selected area, more details about the use case are available in D5.5.

2.5.1 Overview

The CPW use case addresses the domain of Climate Security in order to provide ad-hoc products supporting decision-makers. Understanding the Climate Security nexus is complex and heterogeneous data are needed to address the different scenarios.

The support to three steps in the GEOINT production are envisioned for the demo (See Figure 27). In a first step, wide subcontinental automated continuous monitoring at coarse resolution is implemented. In specific regions where more details are needed, a second step provides semi-automatic approaches to providing GEOINT analysts with standardized sets of data (across different dimensions) and ad-hoc procedures or analyses are used to gain situational awareness at medium-high resolution. Finally, on a third step, finer resolution data is used to perform ad-hoc analyses, which support the generation of change maps, impact assessments or contingency plans.
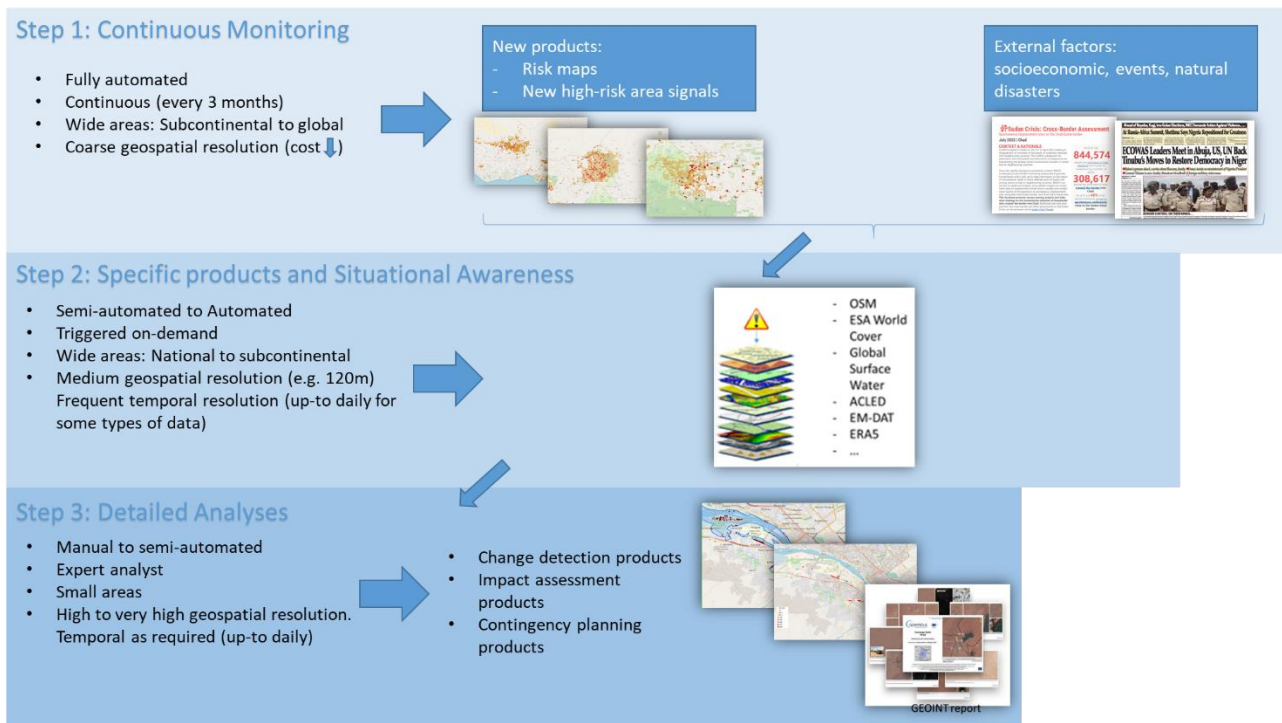
Figure 27: High level representation of the approach for the demo

In the **first step**, a wide sub-continental area (Sahel) is monitored at a coarse spatial resolution to generate updated risk maps every three months for the period 2017-2022 (the period has been defined based on the availability of all the required data for the model application. A Gradient Boosting Classifier was trained (See 2.5.3) from a homogenised stack of heterogeneous data, including socioeconomic data (e.g. accessibility to urban areas, night-time lights, distribution of ethnicities), meteorological data (e.g. precipitation, temperature averages and indices), EO data and derived indicators (e.g. NDVI), data on natural disasters and information

about the topography are used to classify areas with high risk of being affected by armed conflicts in the subsequent quarter (3 months window). This high-level risk map provides an overview for the whole region, including also derivative layers on "new high-risk areas". This approach is only possible through continuous monitoring approaches such as those implemented in GEM.

The **second step** of the demo focused on an activation in Niger. Quite often, a deeper situational awareness assessment of specific countries or areas is needed by GEOINT analysts. In this case, the traditional approach involves a GEOINT analyst using his/her knowledge and experience to gather data and information over the area of interest at a higher detail level. Such data may allow to identify the co-occurrence of possible triggers of conflicts (e.g. natural hazards, coups, presence of terrorist groups). This process is very specialized, often performed manual (hence time-consuming), it is difficult to keep provenance of data (which is essential for trustworthiness of the analysis) and the results may vary depending on the expertise of the analyst.

The developed procedure aims therefore to standardize the process reducing based on selected datasets. From the high level analysis, Niger showed a fragile economy (D5.5, Section 4.1). It is also a region severely affected by both floods and drought issues (D5.5, Section 4.2) and with a high concentration of conflicts (D5.5, Section 4.3). In fact, Niger faced a difficult year of conflict in 2021, with civilian deaths reaching an all-time high (as reported by ACLED[11]). Influence of extremist groups was on the rise, with the Greater Sahara branch of the Islamic State West Africa Province (ISWAP-GS) behind almost 80% of total civilian fatalities in 2021's in Niger, which triggered the creation of local defence groups in the Tillaberi and Tahoua areas and numerous clashes and fatalities. In Tillaberi region, also JNIM (Jama'at Nasr al-Islam wal Muslimin's) was steadily moving closer to Niamey, with violent incidents recorded within 30 km of the capital. The rising violence and community armament has been reported to increase the risk of larger communal conflict by different organisations such as ACLED.

Recently, the military takeover in Niger (which is the ninth in a long series of coups and coup attempts that have destabilised West and Central Africa over the past three years) raises serious concerns for the future of the Sahel. The Sahel has become the epicentre of terrorism globally, accounting for 43 percent of terrorism deaths in 2022, more than both South-Asia, the Middle East and North Africa combined. The situation of Niger, which stood out as "last bastion of democracy in the Sahel," and it was moreover considered the "last bulwark against jihadist and Russian influence" across the region may certainly pose serious risks for the region, impacting counter-terrorism efforts, potentially benefitting terrorist organisations, and eventually impacting the broader Sahel's security landscape[12].

A detailed analysis has been performed exploiting the new service-prototype enabling the discovery, access, extraction and integration of heterogeneous data sources (2019-2022). In this step, in terms of EO and meteo data, the GEM datacubes at 120m resolution have been employed (See D5.5).

Finally, as a **third step** a flood event is analysed at finer resolution. In this case, the end user is not interested in a wide area analysis, but in a detailed analysis of a local area affected by the hazard. The developed tools and methodology are applied to produce ad-hoc analysis and impact assessment considering the support of VHR data, which provides very useful information for analysis at local scale.

---

[11] https://acleddata.com/10-conflicts-to-worry-about-in-2022/sahel/
[12] https://www.icct.nl/publication/unravelling-niger-coup-and-its-implications-violent-extremism-sahel

2.5.2 Scope Area

The scope area for the demo is the Sahel region. More details about this area, consistently affected by drought and considered by experts as a climate change hotspot, can be found in D5.5. In particular, the full Sahel is covered in the continuous monitoring of conflict risk at a resolution of 0.1º. Then, an activation was executed in Niger and a flood event was detected an analysed in three cities.

AOI	AOI Area (Km2)	Resolution	Period
Sahel	13.5 million	0.1º	2017-2022
Niger	1.267 million	120 m	2019-2022
Ad-hoc regions	3 regions selected (between 100 and 400 km2)	10m – <1 m	June-November 2020
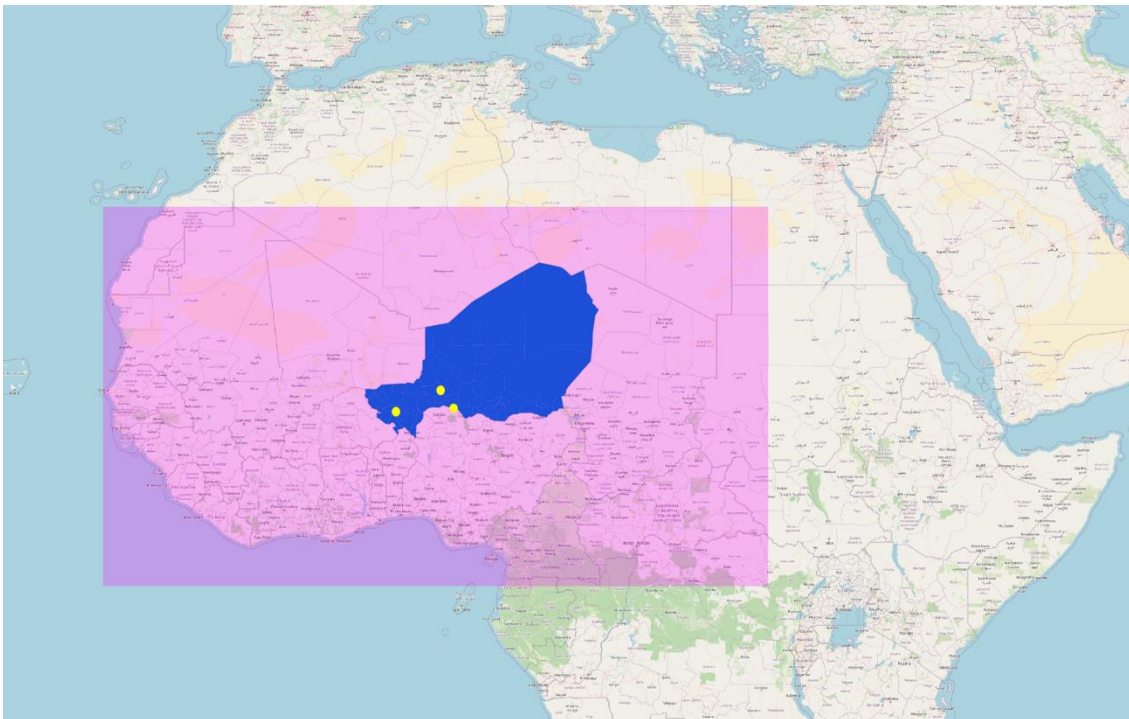
Figure 28. AOI used in demo. In pink, Sahel. In blue, Niger. Yellow points, Niger cities where impact of flooding was analysed.

2.5.3 Model Type & Parameters

In the first step, a homogenised stack has been built with multiple open sources of information described in the table below. This dataset has been used to train, over the wide area, an Extreme Gradient Boosting (XGB) regression trees classifier. The classifier has been trained to predict high-risk conflict areas in a 0.1ºx0.1º grid.

To train the model, the data on conflicts has been used to continuously generate yearly maps of risk over a rolling window of three months, by assigning a value of 1 for the grid cells where conflict happened throughout the year (high-risk) and a value of 0 to the grid cells without conflict over the year (low-risk). For the training phase, an equal number of positive and negative samples have been randomly selected to train the model.

The model has been continuously re-trained with new samples: The first model was trained on data from 1 year of data, finalizing in 2017 Q1 (meaning from 1st April 2016 until 31st March 2017). The classifier metric chosen is the AUC and the parameters used are listed below:

params = {'objective': 'binary:logistic', 'eval_metric': 'auc', 'colsample_bytree': 0.3, 'learning_rate': 0.01, 'max_depth': 5, 'alpha': 10, 'n_estimators': 100, 'num_class' : 2 }

Then, the model was used to generate a risk map in the area for the subsequent time window, e.g., 2017 Q2 (meaning risk of conflict until 30th Jun 2017) and validated against the conflicts reported in the same timeframe. For subsequent iterations, the previous model was again retrained with the data from the most recent year (12 months) in a 3-month sliding window approach.

For the situational awareness phase of the demo, some of the methods to analyse data described in detail in D5.5 were used. In particular, the more relevant are:

- Socio economic data gathering for high-level contextual awareness. Violin plots in combination with time series were used for the analysis.

- Filtering, aggregation and correlation of conflict and disaster datasets exploited through Elasticsearch cluster and Kibana dashboard. The data includes relevant information about actors involved in conflicts such as extremist groups. Its aggregation capabilities also support the identification of proliferation patterns in certain areas.

- Homogenization of ancillary data (e.g., population, ethnicities, urban accessibility...).

- Historical water detection and flooding analysis. As many reports on natural disasters (e.g., floods or droughts) are not detailed in the geospatial domain and typically report floodings at the country or region level, specific assessment has been done to assess at higher granularity where and when floods tend to happen, as different regions are more prone to flooding in different seasons (rainy or dry seasons).
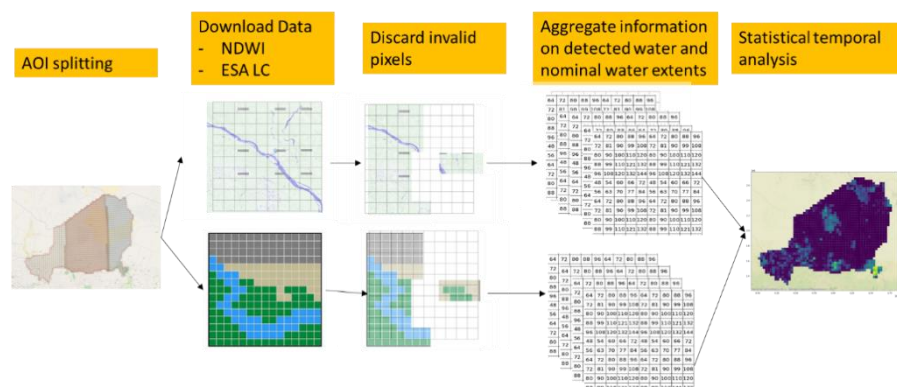
- Anomaly detection of EO and meteo data, by comparing the historical analysis with forecast data. This allows identifying situations in which precipitations or temperature patterns go above/below what is normal for the same period of the year across the historical data.

2.5.4 Input data

A summary of the input data is provided in the Table 2.11 below. In the comments section, it is specified which data are used in the three steps (1. Low resolution risk modelling, 2. Medium Resolution Situational Awareness, 3. High Resolution Detailed Analysis). More details on the data are available in D5.5.

Table 2.11: Summary of the input data for conflict pre-warning use-case

Category	Series Name	Data Source	Temp & Spatial resolution	Temporal availability	Comments
Migration	Internally displaced persons, total displaced by conflict and violence (number of people)	WoldBank - IDMC	Yearly Country-scale	2009-2021	Step 2
Socio-economic	GDP, population growth, inflation, unemployment, mortality rates, access to water, CPIA transparency, accountability, and corruption in the public sector rating, etc.	WorldBank	Yearly Country-scale	1960-2021	Step 2
Socio-economic	World Population High Density Maps	WorldPopHub	Yearly 1km	2000-2022	Step 2
Disasters	EM-DAT	EM-DAT	Daily Sub national	1900-2022	Step1, 2
Disasters	Sendai DesInventar	DesInventar Sendai	Daily Sub national	2015-2022	Step 2
Ethnicities	Ethnic Diversity	GeoEPR	- Sub-national	1960-2021	Step 1, 2
Earth Observation	Sentinel-1, Sentinel-2	Sentinel-2	~5 days Country-scale	2016-2023	Step 2, 3
Earth Observation	Average NDVI	MODIS- MOD13C1	16 day composite 0.05 degree	2000-2023	Step 1
Earth Observation	ESA World Cover	ESA WC dataset	Yearly 10m	2020-2021	Step 2, 3
Earth Observation	Night-time lights	VIIRS	Daily 370m – 740m	2011-2022	Step 1, 2
Earth Observation	Topography (DEM)	-	Static SRTM 3 sec	Static	Step 1, 2, 3
Meteorological	meteoblue datasets API - Temperature - Precipitations - Soil Moisture	meteoblue	Hourly/Daily NEMSGLOBAL (30 km)	1990-2021	Step 2

Meteorological	Precipitations and temperature	Climate Research Unit TS4.0 global dataset	Monthly	1901-2022	Step 1
Conflict	ACLED Events	ACLED	Daily Sub national (e.g. town, part of region)	1996-2021	Step 1, 2
Other	Urban accessibility	GAM JRC	Static 30 arc seconds	2000/2015	Step 1
Other	OpenStreetMap	OpenStreetMap	Static with regular updates	Static with regular updates	Step 3

2.5.5 Results

2.5.5.1 Step 1 - Risk maps (every three months in period 2017-2022)

As introduced in Section 2.5.3, the model is trained and validated with data on the corresponding time frame (e.g. 2017Q1), while the test metrics reported below correspond to the performance of the model in support of an early-warning system, i.e. model trained in 2017Q1 tested to forecast conflict areas for 2017Q2 and scores computed once the data of 2017Q2 is available.

In terms of validation, the average AUC for the period of analysis (2017-2022) is 0.805. The minimum 0.78 and the maximum 0.84, which could be considered valid results.
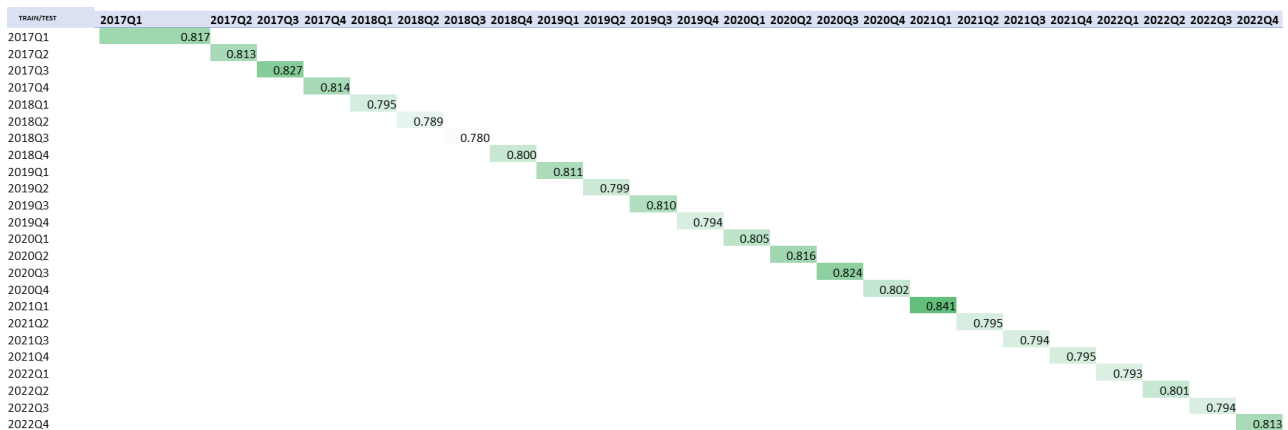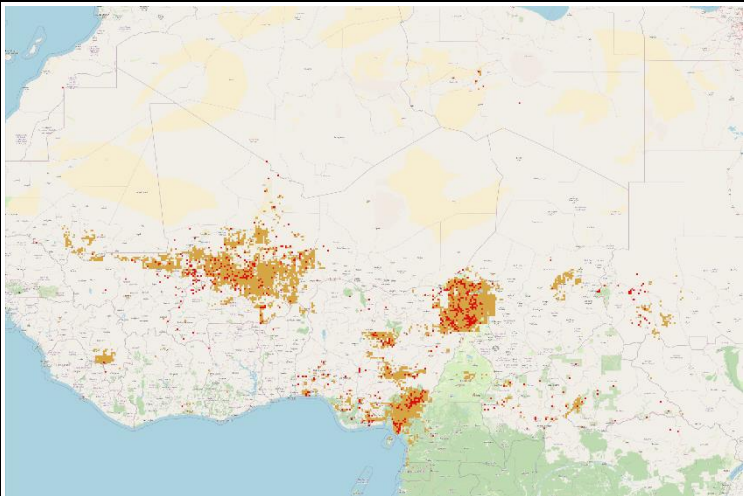
Regarding feature importance, the top5 ranking of features by decreasing importance is:

1. NDVI (2.533)
2. Standardized Precipitation Index, 24 months (2.498)
3. Mean precipitations (2.360)
4. Standardized Precipitation Index, 12 months (2.256)
5. Standardized Temperature index, 24 months (2.175)

Regarding the test metrics, the average AUC is 0.758, minimum (cumulative training until 2022 Q2, prediction for 2022 Q3) 0.62 and maximum (cumulative training until 2020 Q4, prediction for 2021 Q1) 0.834.

TRAIN→ TEST↓	2017Q1	2017Q2	2017Q3	2017Q4	2018Q1	2018Q2	2018Q3	2018Q4	2019Q1	2019Q2	2019Q3	2019Q4	2020Q1	2020Q2	2020Q3	2020Q4	2021Q1	2021Q2	2021Q3	2021Q4	2022Q1	2022Q2	2022Q3
2017Q2	0.790																						
2017Q3		0.759																					
2017Q4			0.737																				
2018Q1				0.773																			
2018Q2					0.799																		
2018Q3						0.754																	
2018Q4							0.773																
2019Q1								0.810															
2019Q2									0.783														
2019Q3										0.787													
2019Q4											0.708												
2020Q1												0.776											
2020Q2													0.744										
2020Q3														0.750									
2020Q4															0.759								
2021Q1																0.834							
2021Q2																	0.802						
2021Q3																		0.628					
2021Q4																			0.754				
2022Q1																				0.803			
2022Q2																					0.827		
2022Q3																						0.620	
2022Q4																							0.664

The risk map for the best performing forecast model is discussed below.

Overview Map	Analysis
  • **Orange pixels represent predicted high-risk areas** • **Red pixels represent the presence of conflict for the time period**	The model properly predicts the main conflict areas for the period, such as the Lake Chad Basin, the borders between Mali-Burkina Faso and Niger, and between Nigeria and Cameroon. The high-risk areas predicted by the model are wider than the resolution (0.1º grid) used to accumulate the presence of conflict. This seems reasonable as the variables analysed do not vary in space so rapidly, hence the model is not able to predict in high level of detail where exactly the risk of conflict will happen, but it is able to pinpoint coarsely the areas where conflicts are likely to happen

Close-up views
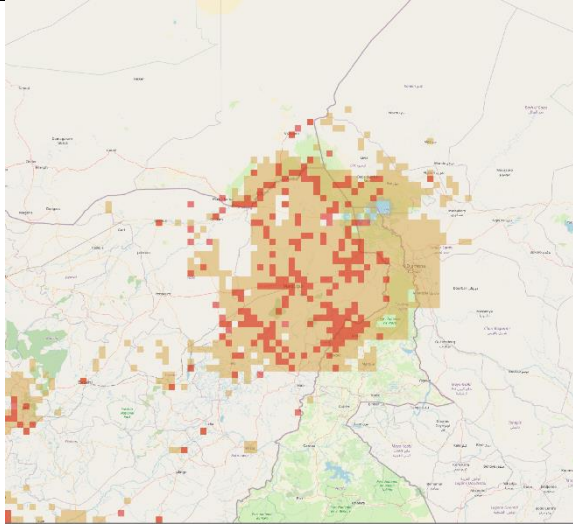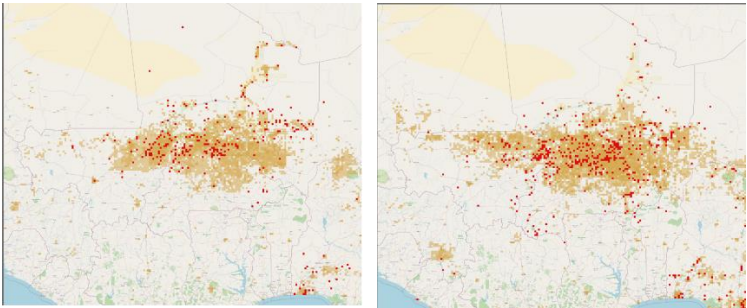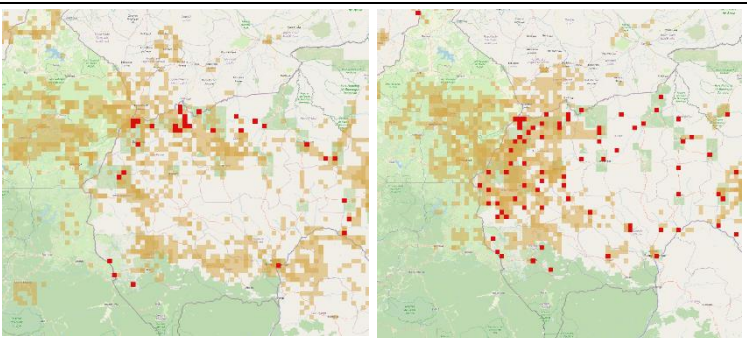
Lake Chad Basin	Nigeria-Cameroon Border
Prediction of high-risk conflict areas for 2021Q1 in Lake Chad Basin Area	Prediction of high-risk conflict areas for 2021Q1 in Nigeria-Cameroon borders
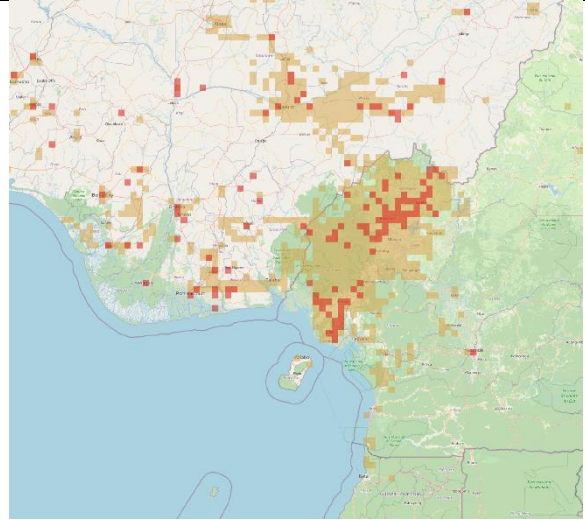
Temporal Evolution

Accumulated predictions of conflict areas (orange) and actual conflicts (red) in the Mali-Burkina Faso-Niger borders for 2018 (left) and 2021 (right)

Mali - Burkina Faso – Niger borders

The borders between Niger and Burkina Faso suffered an increase in conflicts throughout 2019, 2020 and 2021. This effect is captured by the model, that enlarges the high-risk area, highlighting the increased risk.

Accumulated predictions of conflict areas (orange) and actual conflicts (red) in the Cameroon-CAF borders for 2018 (left) and 2021 (right)

Cameroon – Central African Republic.

The number of conflicts in the area grew from around 140 conflict events in 2018 to more than 400 conflict events in 2021. The model showcases more high-risk pixels in the area of the border for 2021 than it does for previous years, following this new trend.

The summary map below highlights the conflict areas aggregated by year, which is aligned with the analysis in the tables above, where the predictions analyzed for 2021 highlight the decrease around the Lake Chad Basin, and the increase in the Mali-Burkina Faso – Niger borders.

2019	1526 (-)	1020 (-)	2497 (-)
2020	1956 (+28%)	1478 (+ 44%)	3801 (+ 52%)
2021	2661 (+36%)	1065 (- 28%)	3860 (+ 2%)
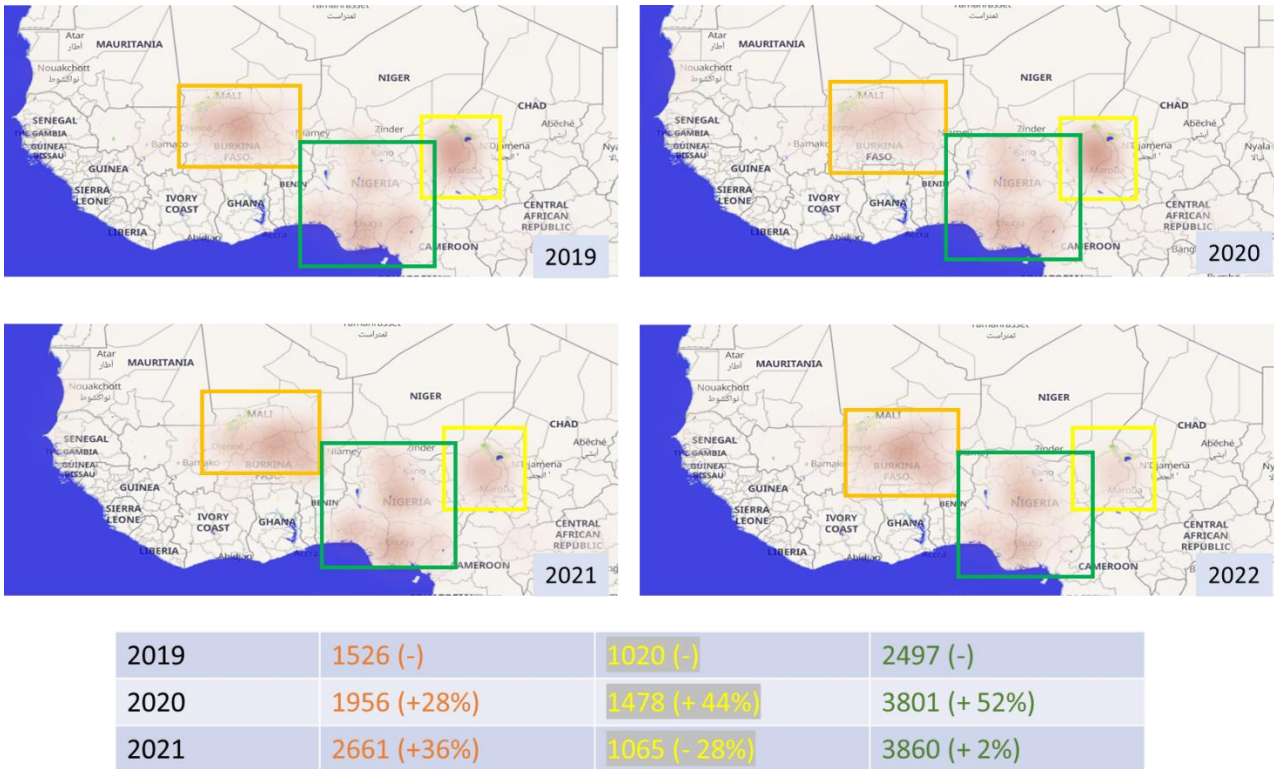
Figure 29: Conflicts evolution in the main conflict clusters to visualize the dynamics of conflict hotspots

2.5.5.2 Step 2 - Niger situational awareness (2019-2022)

Situational awareness in the context of GEOINT (Geospatial Intelligence) production refers to the understanding and perception of environmental elements, events, and dynamics both in time and space, the comprehension of their meaning, and the projection of their future status. It's a critical component for informed decision-making, especially in defence, intelligence, disaster response, and security operations.

Different datasets and products are produced thanks to the GEM project aiming at 1) collecting and integrating geospatial data from various sources, 2) using the gathered data, analysts provide insights into current events, revealing patterns and relationships that might not be immediately obvious, and 3) providing standardized data to homogenise and keep track of the provenance of data used for the GEOINT production.

Different data (See D5.5) were integrated (See Section 2.5.3), providing information on different aspects related to the Climate Security Nexus.

Socio-economic data such as the examples in Figure 9 highlights Niger issues related to water stress, especially in recent dates, while Figure 10 shows the distribution of population in Niger, which is essential to analyse the impact of certain events such as natural disasters in local populations.
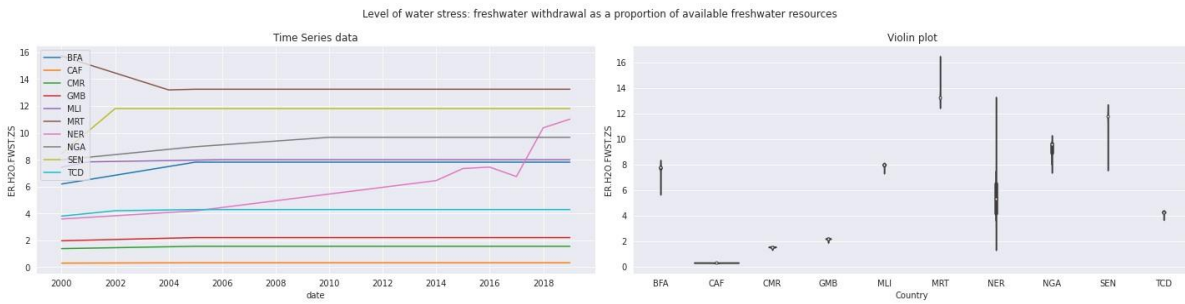
Figure 30: Level of water stress
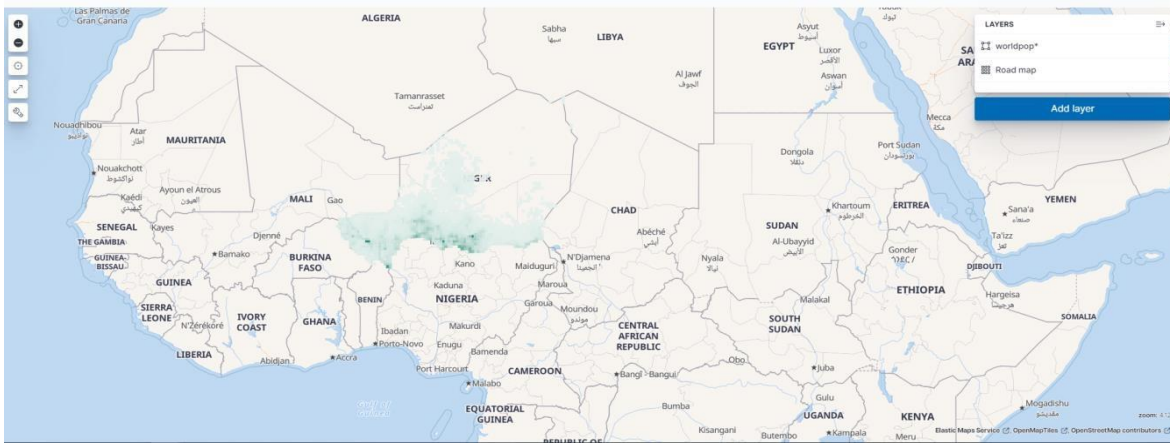
Figure 31: High density population distribution map for Niger

A dedicated processing has been developed to gather Sentinel-2 data at 120m resolution to compute the NDVI and extract a water mask. This water mask is then compared with a reference layer (ESA World Cover) to provide different analyses. Analysing the time series is then possible to identify patterns (e.g., seasonality) and derive risk maps for flooding (See Figure 32). This data is then manually cross checked against online reports over the area. This data is then useful for analysts to better understand potential issues triggering flooding (e.g., displacements or resource scarcity influenced by natural disasters such as floods).
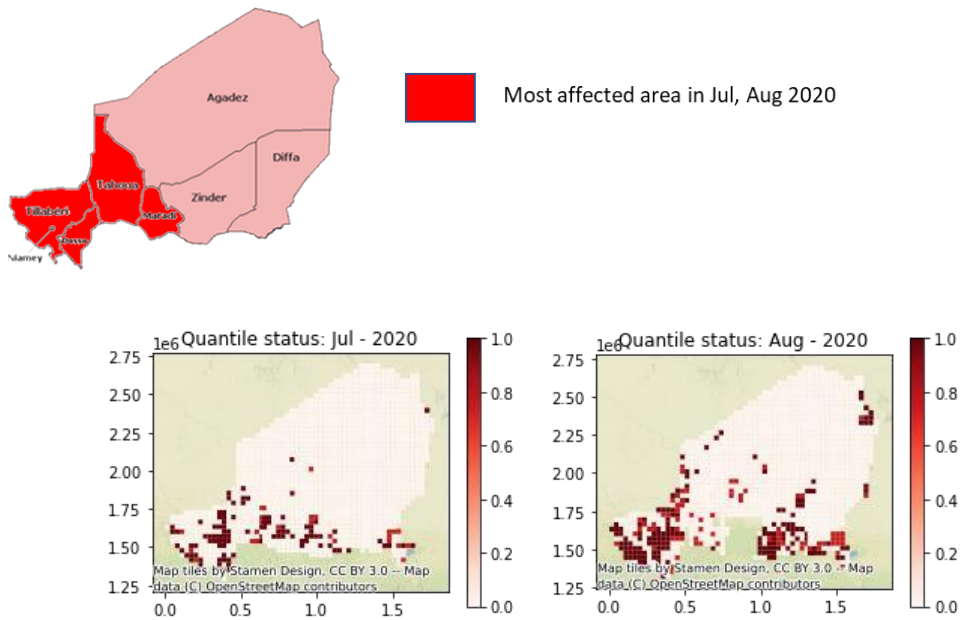
Figure 32: Example of anomalies map (risk of flooding) for 2020

Contextual information is also provided from meteorological data such as the maps shown in Figure 33. Using that data and analysing long time series, a methodology was defined to compare history of precipitations and temperature with short-term forecasts (e.g., around 5 days in the future) providing a risk signal in case of anomalous temperatures or precipitations patterns.
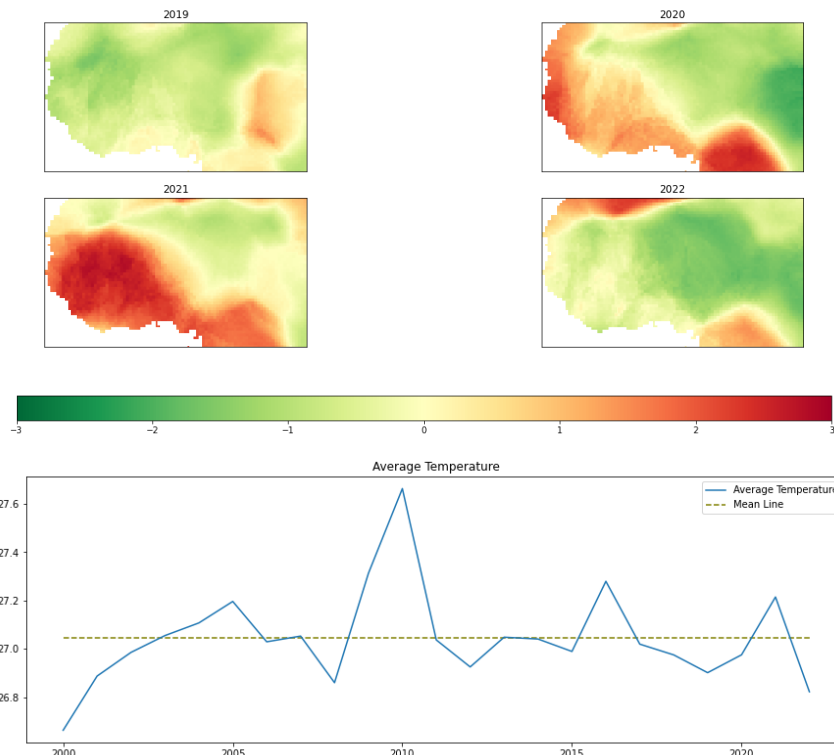
Figure 33: Top: Deviation from yearly average temperature (number of standard deviations). Bottom: Yearly average temperatures

2.5.5.3 Flood in Niamey September 2020: impact assessment

In the third step, it was carried out a detailed analysis of the impact of flooding that took place on Niamey and other cities of Niger in September 2020.

In Summer 2020, Niger suffered days of torrential rainfall that caused the River Niger to burst its banks. The rainfall hit the country since July but in September the flooding caused by River Niger (and others) affected more than 200000 people and killed at least 45 people. Local authorities reported almost 30,000 houses being affected and also the damage to granaries and crop fields.

In the demo, the focus for the impact assessment has been put in Niamey and the cities of Bangi and Tahoua.

Figure 34. Cities selected for the impact assessment.

Using continuous revisit of Sentinel-1 and its independency of weather conditions, it has been computed the evolution of flood areas in Niamey. For that, a thresholding algorithm was applied at 10 meters resolution. The peak of the flooding occurred on 15th September, being visible the consequences till months after, as shown in Figure 35. Despite the precipitations during the months of June, July, August there was no flooding when comparing with previous year, but in September the number of pixels affected reached its peak.
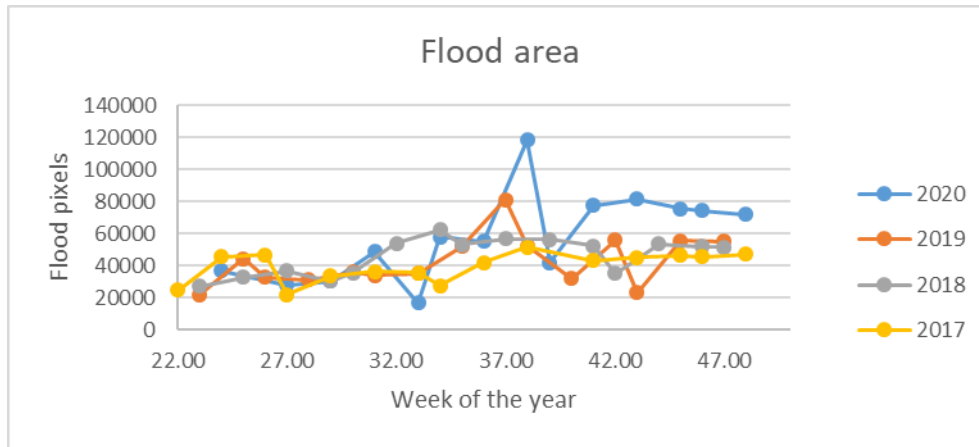
Figure 35. Flood area in AOI from June to November 2017-2020

The Sentinel-1 and Sentinel-2 images on 15th September show clearly the affected area (Figure 36Figure 35) and allows the generation of a flood mask (Figure 37)
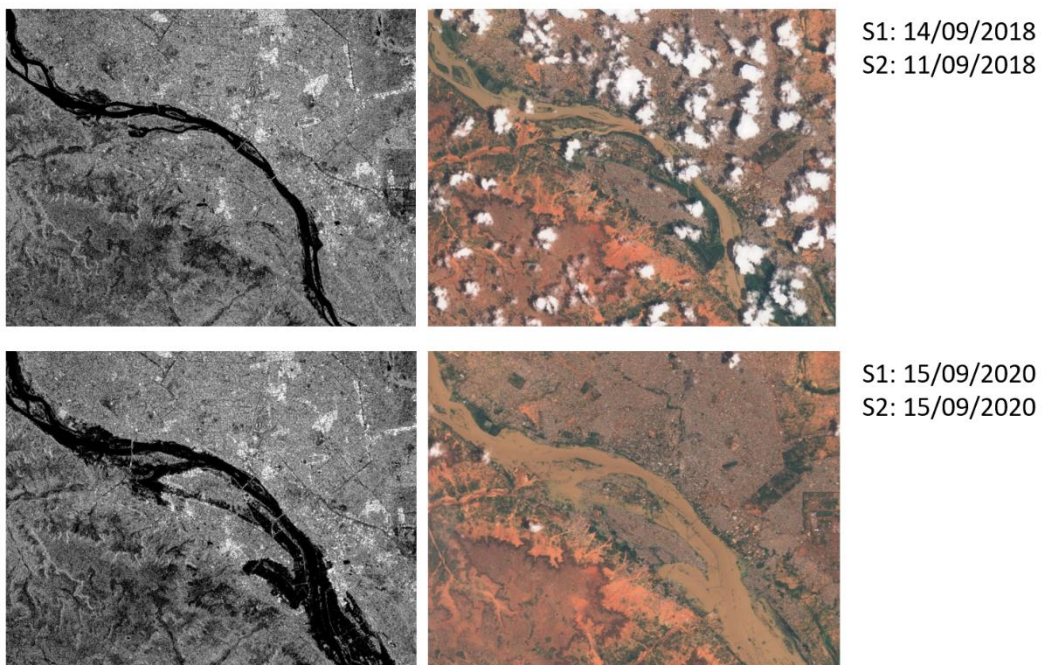
Figure 36. Sentinel images in a reference date and on 15th September 2020 (peak of flooding)

In order to get more information of affected areas, additional datasets have been used to analyse the impact on infrastructure and population. Regarding the infrastructure, OSM provides information of building and infrastructures. If it is carried out the intersection of flood areas with OSM features, it is possible to estimate the affected elements (Figure 38).
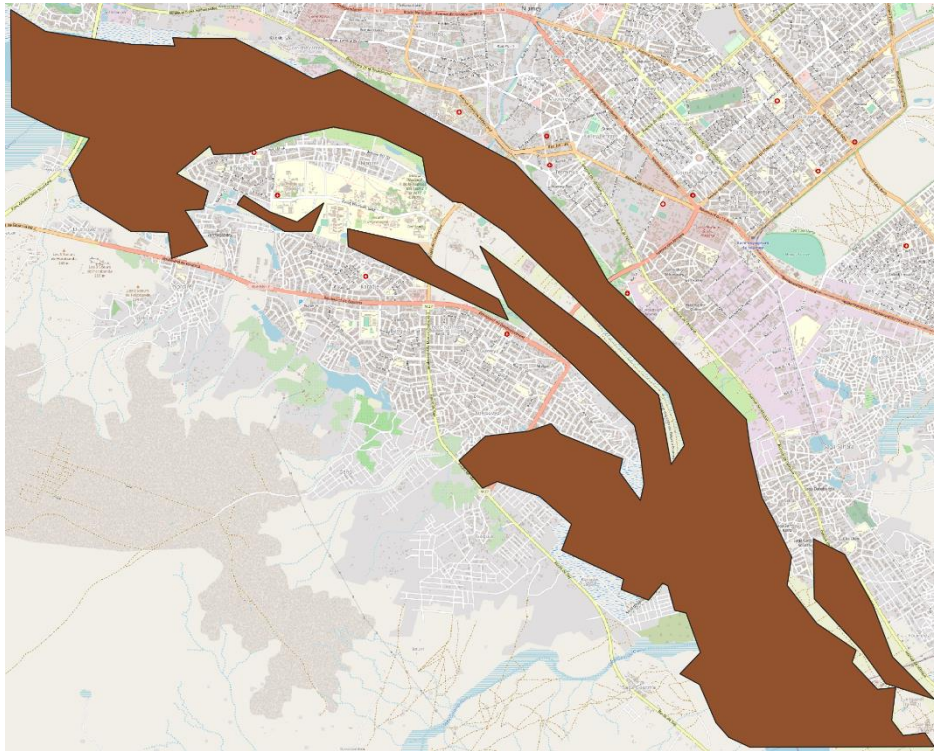
Figure 37. Flood mask in Niamey on 15th September 2020
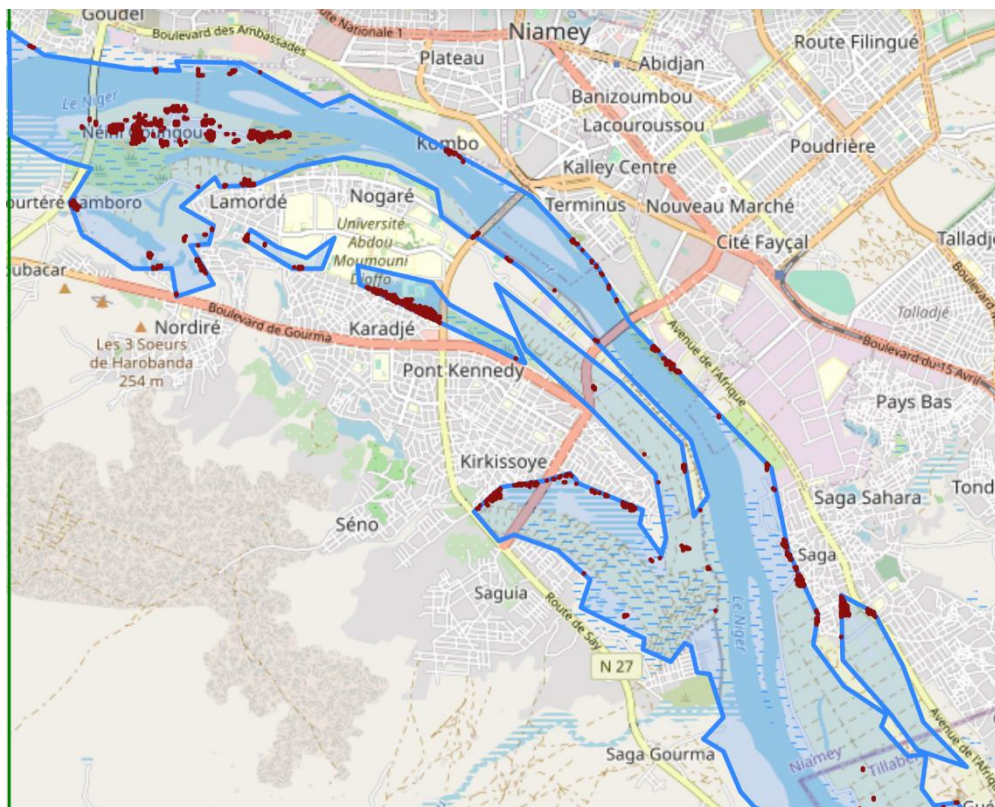
Figure 38. In red, buildings affected by Niamey flooding.

The use of Human Settlement Layer allows also to estimate the number of people affected by the flooding in Niamey at a given date.
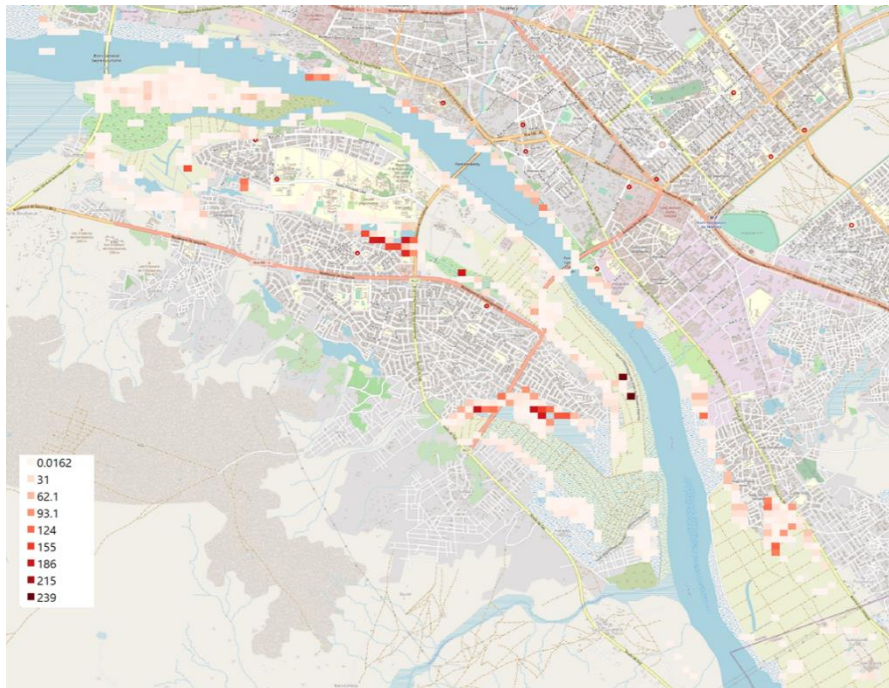
Figure 39. Number of people affected by Niamey flooding.

The use of VHR imagery enables to obtain the flood mask at a better resolution to improve the accuracy of the assessment. The VHR imagery is not acquired regularly: a request needs to be done to image providers. This is possible when the hazard to be monitored can be predicted or when the analysts is interested on monitor a specific area. As an example, after the flooding in Niamey and activation (EMSR 466) was made in other cities to assess the impact.

As performed in EMSR466, using GEM developments it is possible to apply the same methodology in Bangi to determine if the city was affected or not. Also, requesting VHR imagery allows this same analysis to be done at a spatial resolution of 0.5-3 meters.
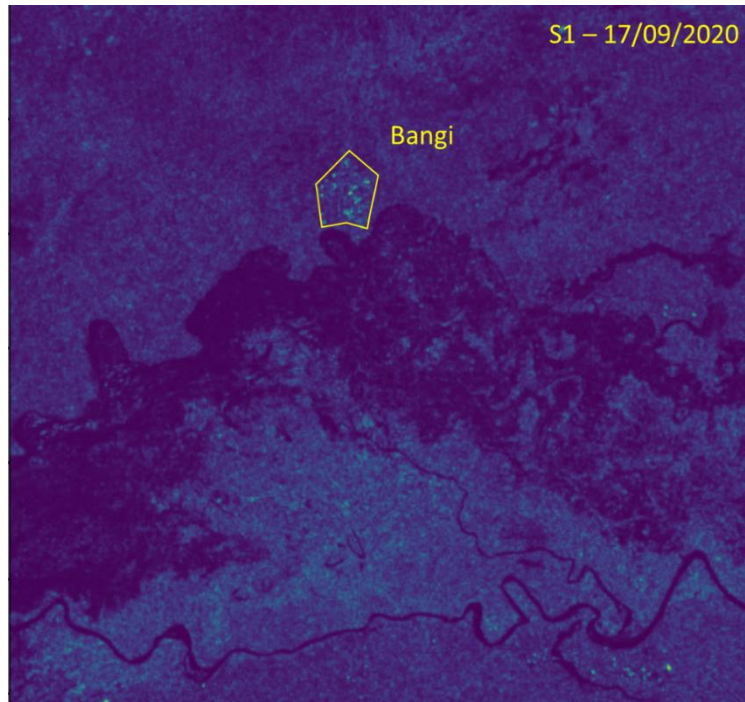
Figure 40. Sentinel-1 Coloured image over Bangi.
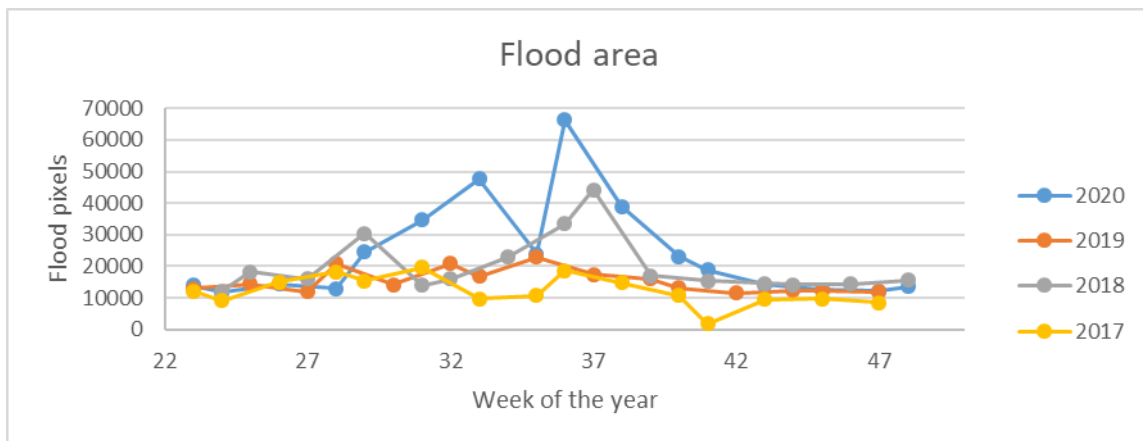
Figure 41. Flood area in Bangi AOI from June to November 2017-2020

In Bangi, the analysis performed shows that there is also flood area reaching its peak on September 2020, but in this case there is no direct impact on people or buildings, since according to ESA World Cover, the affected area is mostly cropland, grassland and shrubland, not built-up areas are affected (Figure 42).
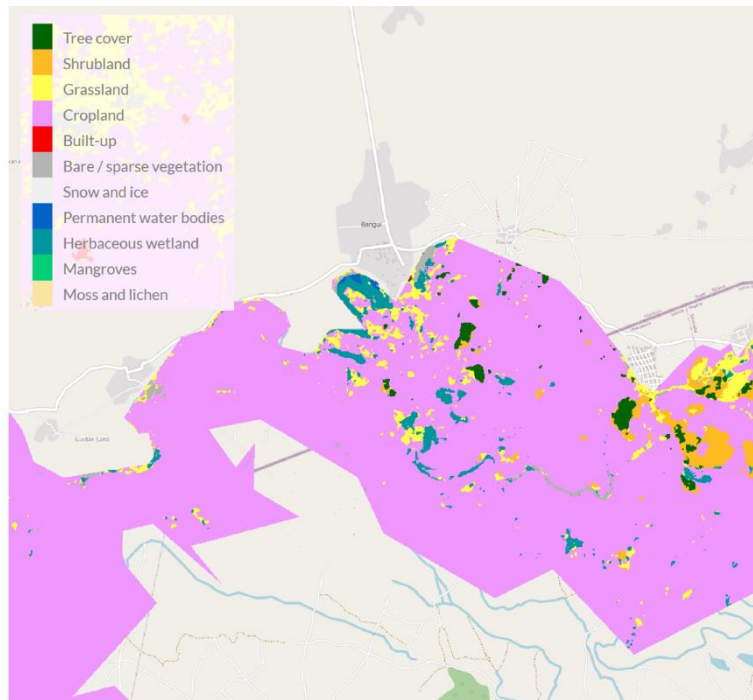
Figure 42. Land cover of flood areas in Bangi AOI.

Finally, for the demo it was carried out also the analysis of the impact in Tahoua using Sentinel-2 at its native resolution of 10m to obtain the flood mask. This is possible only when there are not clouds in the scene. Once the flood mask is available, the impact assessment with complementary datasets can be done in the same way, independently of the origin of the mask (S1, S2, VHR).
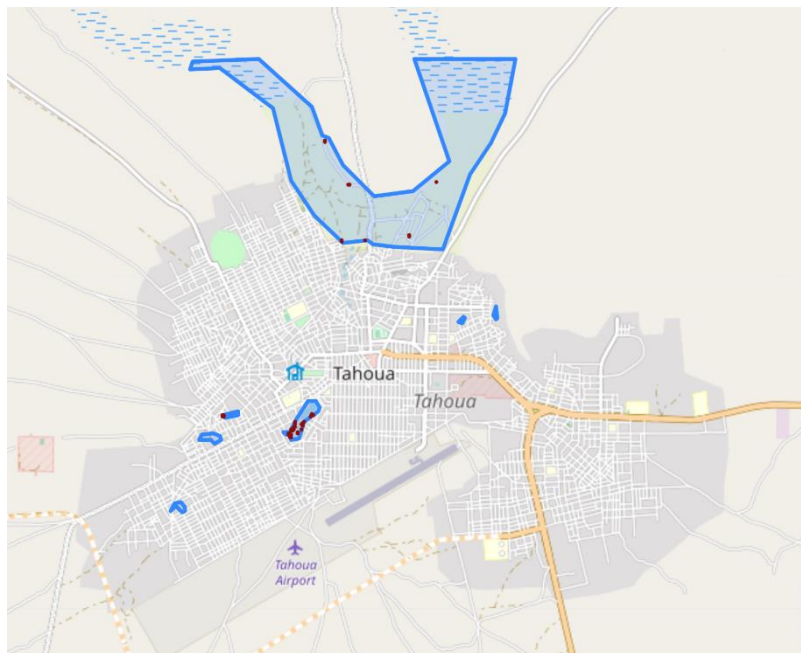
Figure 43. Flood mask (blue) and affected buildings (red) in Tahoua on 27th September 2020.

3 Continuous Monitoring

Continuous monitoring of Earth's environment is crucial for addressing climate change, supporting sustainable socio-economic development, and informing political decisions at local, national, and international levels. It provides the foundation for evidence-based policies and actions that are essential for safeguarding our planet's future and ensuring the well-being of current and future generations.

EO programs like Sentinels or LandSat provide objective view of Earth and play a pivotal role in continuous monitoring due to their capabilities and comprehensive coverage, both spatially as well as temporally. With EO imagery it is possible to monitor changes in natural ecosystems, like deforestation, urban expansion, wetland degradation, glacial retreat, sea-level rise, shifts in vegetation patterns; it is possible to provide insights into crop health, soil moisture, land productivity, track water resources and water quality, urban growth, infrastructure development, deforestation and forest degradation, support biodiversity conservation efforts and many more.

In general, the observations contribute to a better understanding of the Earth's changing climate and help in the development of climate change mitigation and adaptation strategies.

3.1 Normalized Difference Water Index

In GEM, we have set out to show how Continuous Monitoring can be done cost-effectively at scale. This section describes how one can employ GEM framework to continuously monitor Sahel region for anomalies of the normalized difference water index (NDWI).

In an approach, similar as defined for monitoring plant water stress[13], we have set out to monitor water content using the NDWI defined by McFeeters[14]:

Sentinel-2 NDWI = (B03 - B08) / (B03 + B08)

which can be used to monitor changes related to water content in water bodies. As water bodies strongly absorb light in visible to infrared electromagnetic spectrum, NDWI uses green and near infrared bands to highlight water bodies. Index values greater than 0.5 usually correspond to water bodies. Vegetation usually corresponds to much smaller values and built-up areas to values between zero and 0.2.

Additionally, we only calculate NDWI for pixels where there is data available, and the pixel is not marked as cloudy (both available in Sentinel Hub using dataMask[15] and cloudMask[16] "bands").

---

[13] https://edo.jrc.ec.europa.eu/documents/factsheets/factsheet_ndwi.pdf

[14] S. K. McFeeters (1996) The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features, International Journal of Remote Sensing, 17:7, 1425-1432, DOI: 10.1080/01431169608948714

[15] https://docs.sentinel-hub.com/api/latest/user-guides/datamask/

[16] https://docs.sentinel-hub.com/api/latest/user-guides/cloud-masks/

3.2 NDWI baseline statistics

To obtain the baseline statistics, which will be needed to find anomalies, historical data is needed. In our case we calculated the per-month baselines from Sentinel-2 L1C data from January 2020 until December 2022 for each pixel of the AoI. Figure 44 shows how mean and standard deviation of NDWI is changing throughout the year for one EOPatch.
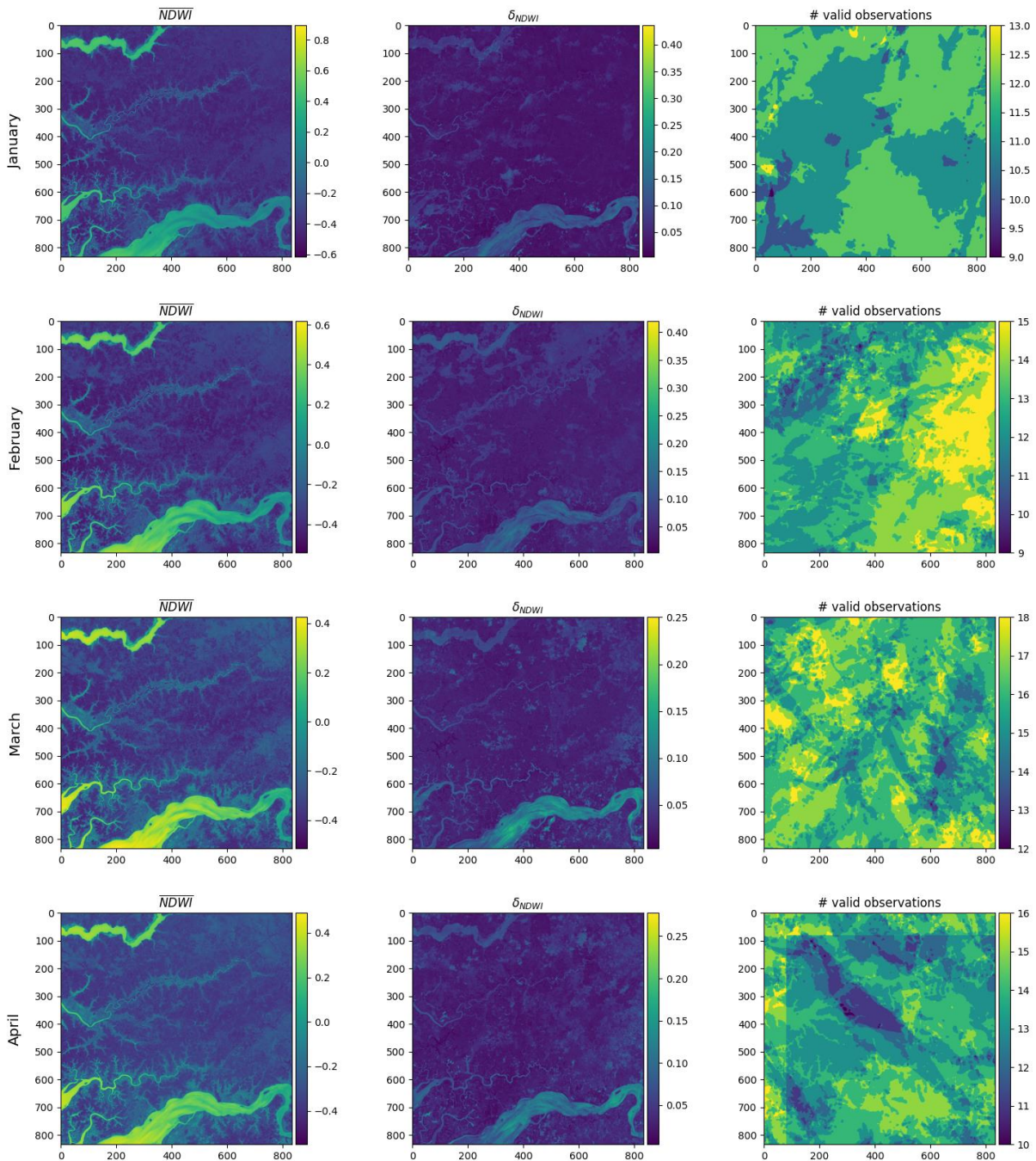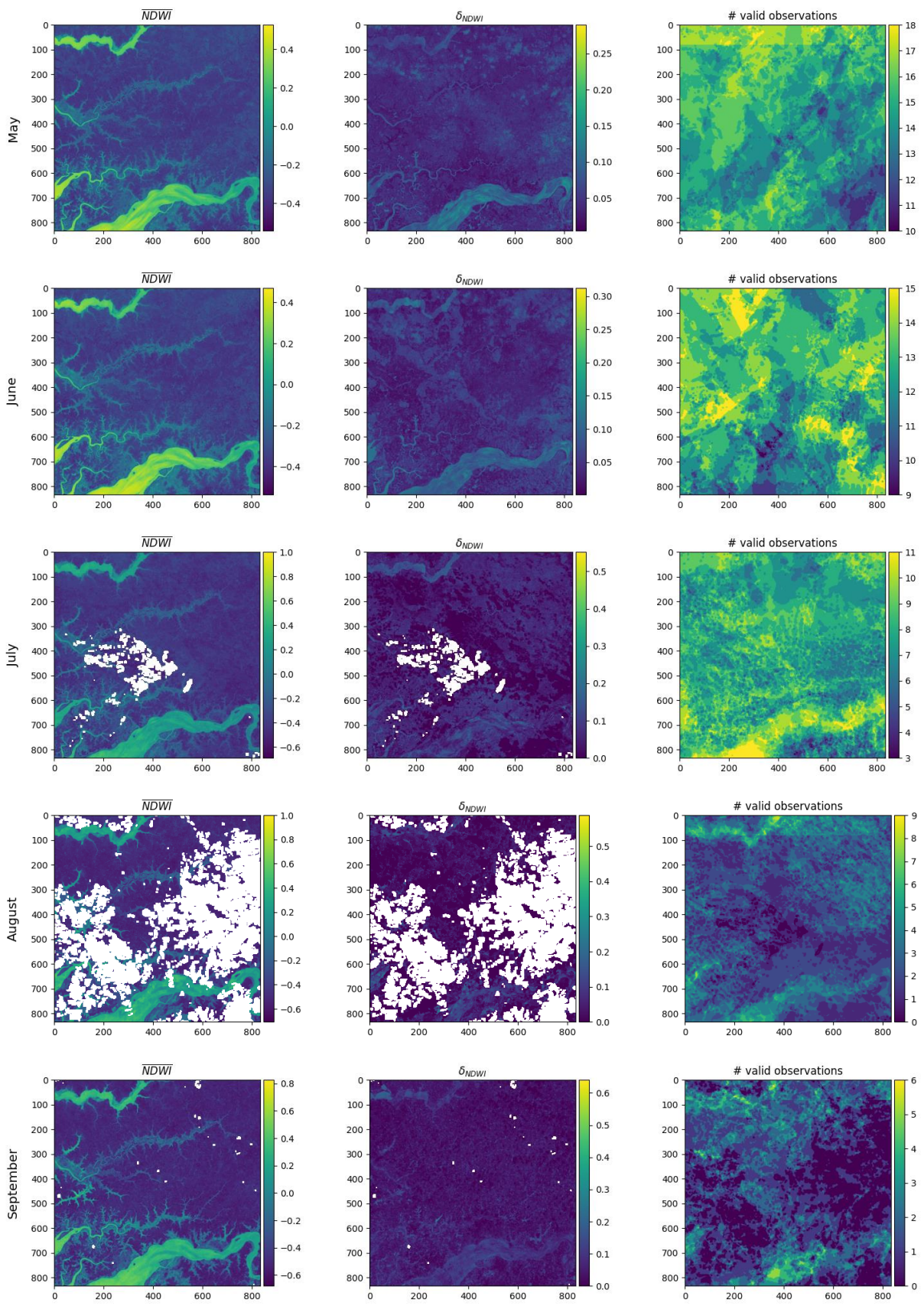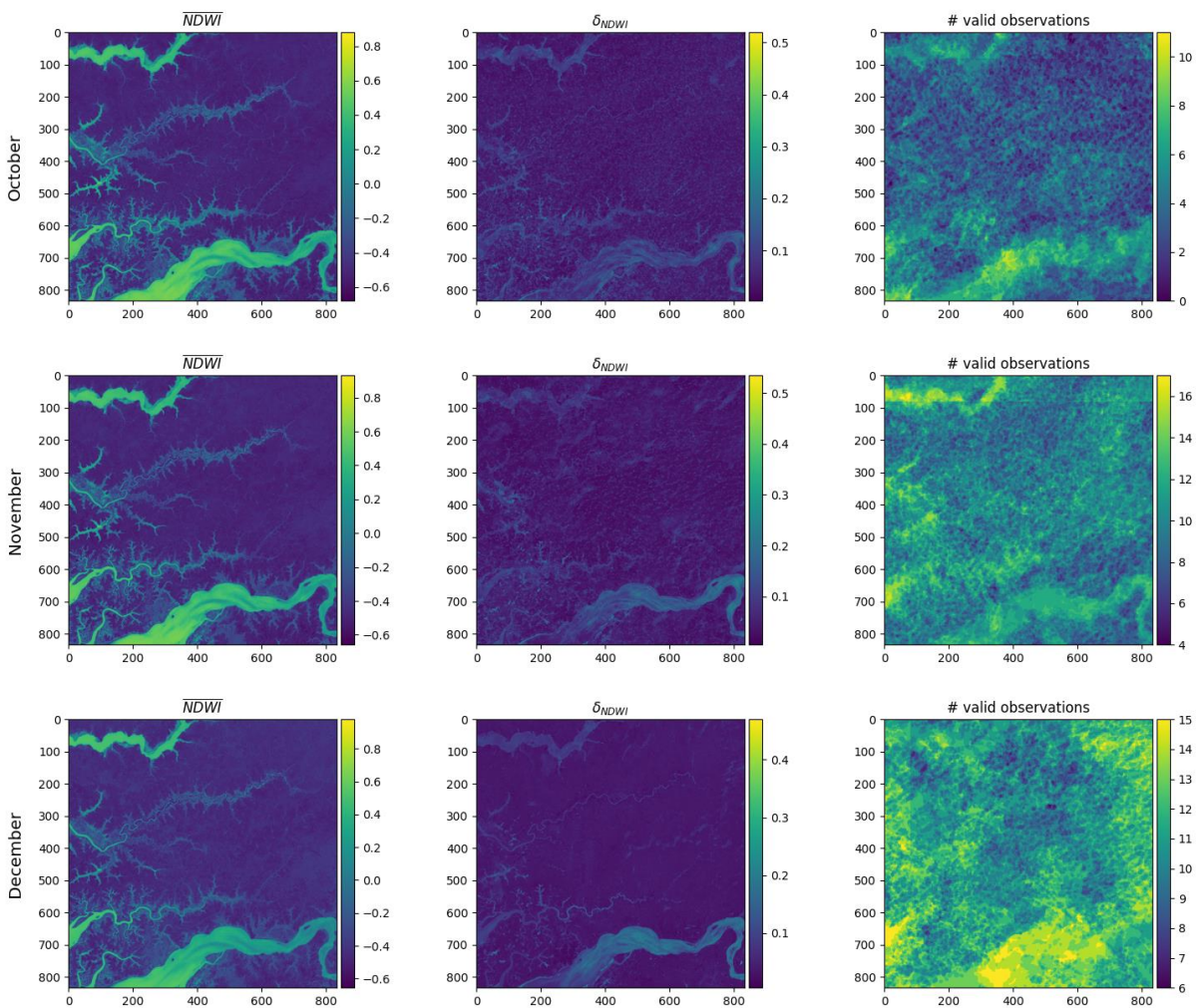
Figure 44: NDWI baseline statistics over months.

Looking at the data from another perspective is shown in Figure 45, where NDWI statistical baseline is plotted for two points – one over a river, where NDWI is high throughout the year, and another over an arid region, where NDWI over the year is low. The disjoint time series for location (B) on image Figure 45 depicts issues visible also in Figure 45 for summer months – some parts do not have enough valid observations over the time period used to build the statistical baseline. We define for each pixel to have a valid observation if it has Sentinel-2 L1C data (which is not true on border of orbits), and it is not cloudy, based on s2cloudless algorithm[17] provided by Sentinel Hub[18].

---

[17] https://github.com/sentinel-hub/sentinel2-cloud-detector
[18] https://docs.sentinel-hub.com/api/latest/user-guides/cloud-masks/
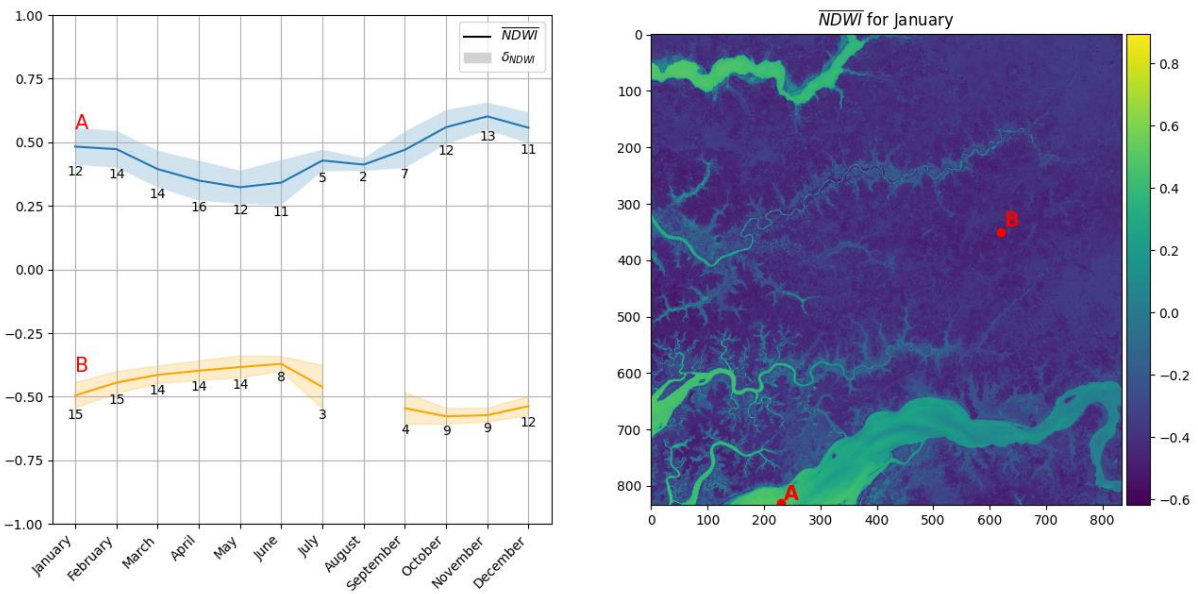
Figure 45: Baseline statistics on a location with high (A) and low (B) NDWI index values. Numbers specify number of valid observations for that particular time interval.

3.3 NDWI anomaly
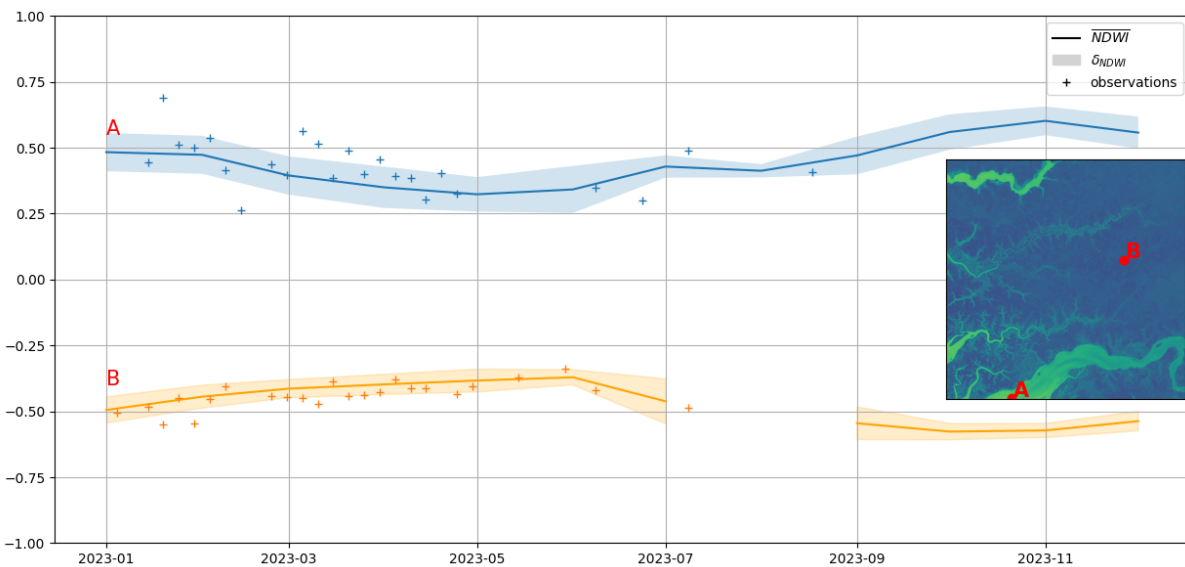
Figure 46: Comparing new observations to historical baseline statistics.

The Figure 46 illustrates how historical baseline statistics can be used to monitor new observations. Each new observation (in Figure 46, the online monitoring period has started in January 2023) can be compared to the baseline – is the new observation within the standard deviation of the average value for that month, or it falls outside the area?

NDWI anomalies are defined with the following approach:

$$NDWI_{anomaly} = \frac{NDWI - \overline{NDWI}}{\sigma_{NDWI}},$$

Where $\overline{NDWI}$ is long-term (historical) mean value of if NDWI over particular period, and σ_{NDWI} is its standard deviation. The NDWI anomaly product is given in standard deviation units. It is commonly ranging from -4 to +4, negative anomalies pointing to lower than expected values (e.g., draught) and positive to higher than expected values (e.g., floods). An example where a Sentinel-2 observation is within the expected values is shown in Figure 47, while Figure 48 shows some larger deviations, but still far from exceeding.
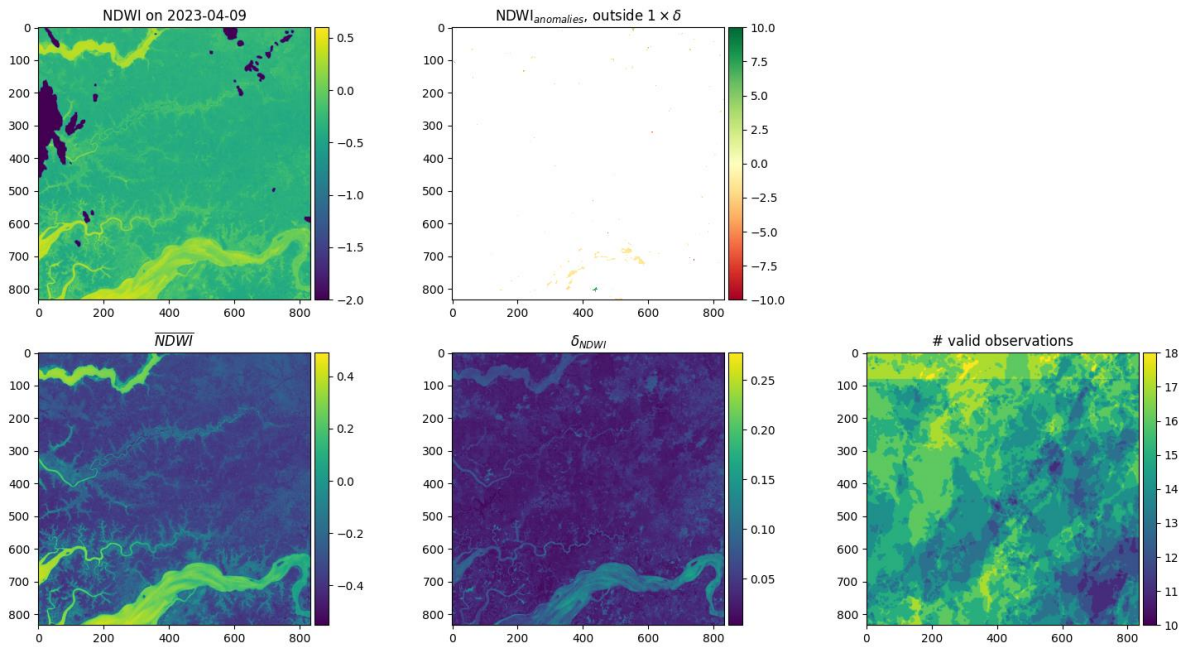
Figure 47: NDWI data for 2023-04-09 is within the expected (one standard deviation) limits.
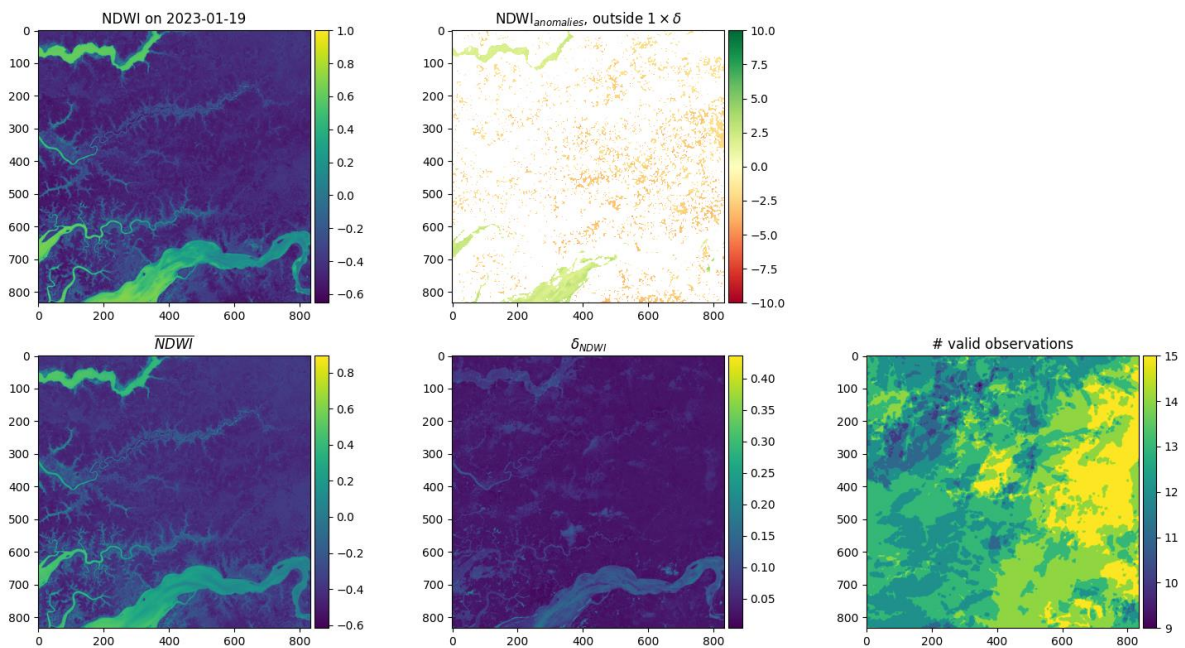
Figure 48: Some bigger deviations observed on 2023-01-19.

3.4 Running service

The main objective of this exercise was to set-up and run the continuous monitoring and demonstrate its capabilities. In this section, we illustrate the main steps of the continuous monitoring performed with GEM framework.

3.4.1 Define AoI

We need the area of interest, typically provided in geojson format. It serves as input to the next step. In our example, we have opted for West Sahel region, shown in Figure 49, roughly 5.5Mio km[2]. As can be seen from Code 1, the geojson with AoI is passed via the geometry_filename parameter. We also see that the configuration specifies resolution to be 120m, and that the area will be split according to Sentinel Hub Batch API[19], using tiling grid[20] with id 2.
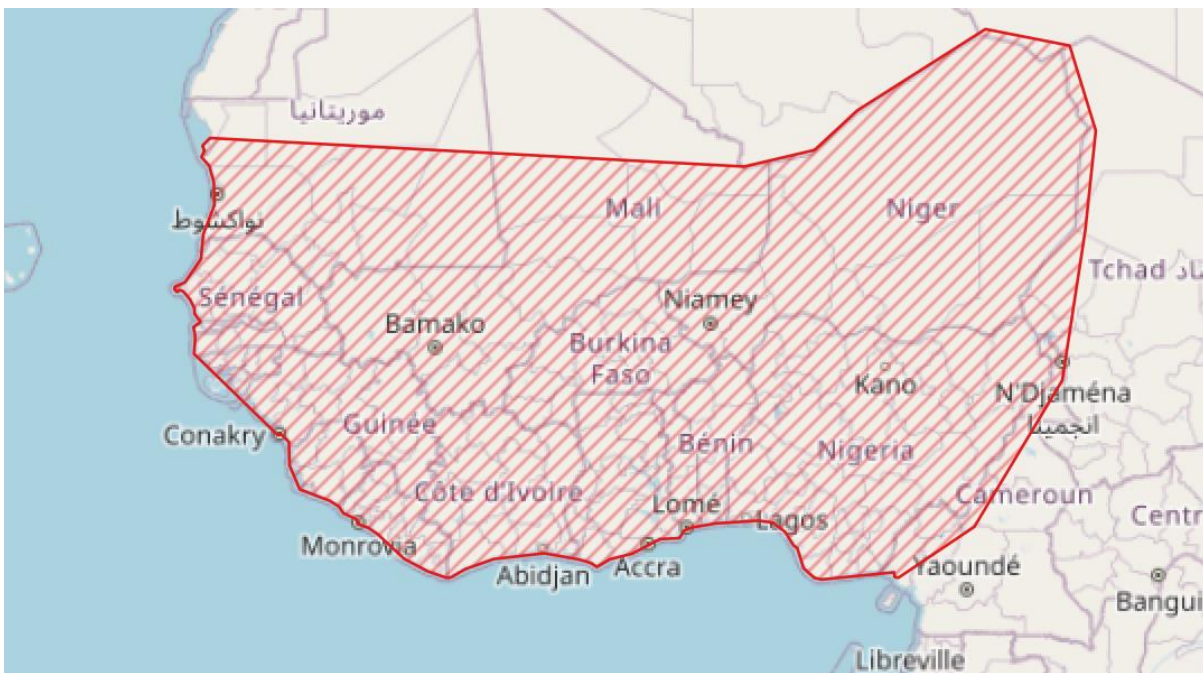
Figure 49: AoI used in the continuous monitoring example.

Code 1: eo-grow configuration for the area.

```
"area": {
    "manager": "eogrow.core.area.BatchAreaManager",
    "geometry_filename": "aoi_region.geojson",
    "tiling_grid_id": 2,
    "resolution": 120
}
```

---

[19] https://docs.sentinel-hub.com/api/latest/api/batch/
[20] https://docs.sentinel-hub.com/api/latest/api/batch/#tiling-grids

3.4.2 Obtaining data

For the AoI, we then obtain the input data – NDWI index from Sentinel-2 L1C data. For data retrieval, we make use of Sentinel Hub Batch API, which is the main component of the datacube engine. We have used the evalscript shown in Code 2 to obtain data for historical baseline statistics.

Code 2: Evalscript for creating datacube with NDWI index, filtering out non valid pixel observations.

```
//VERSION=3

function setup() {
  return {
    input: ["B03", "B08", "CLM", "dataMask"],
    output: [{ id: "NDWI", bands: 1, sampleType: "FLOAT32"}],
    mosaicking: Mosaicking.ORBIT
  };
}
function updateOutput(outputs, collection) {
    Object.values(outputs).forEach((output) => {
        output.bands = collection.scenes.length;
    });
}

function updateOutputMetadata(scenes, inputMetadata, outputMetadata) {
  let timestamps = []
  scenes.forEach(scene => {
    timestamps.push(scene.date);
  })
  outputMetadata.userData = {
    timestamps:  JSON.stringify(timestamps)
  };
}

function is_bad(sample) {
  return ((!sample.dataMask) || sample.CLM)
}

function evaluatePixel(samples) {
  var n_observations = samples.length;
  let ndwi = new Array(n_observations).fill(0);

  samples.forEach((sample, idx) => {
   ndwi[idx] = is_bad(sample) ? -2 : index(sample.B03, sample.B08)
  });

  return { NDWI: ndwi};
}
```

The resulting datacubes were then converted to EOPatches. The grid of EOPatches can be seen in Figure 50.
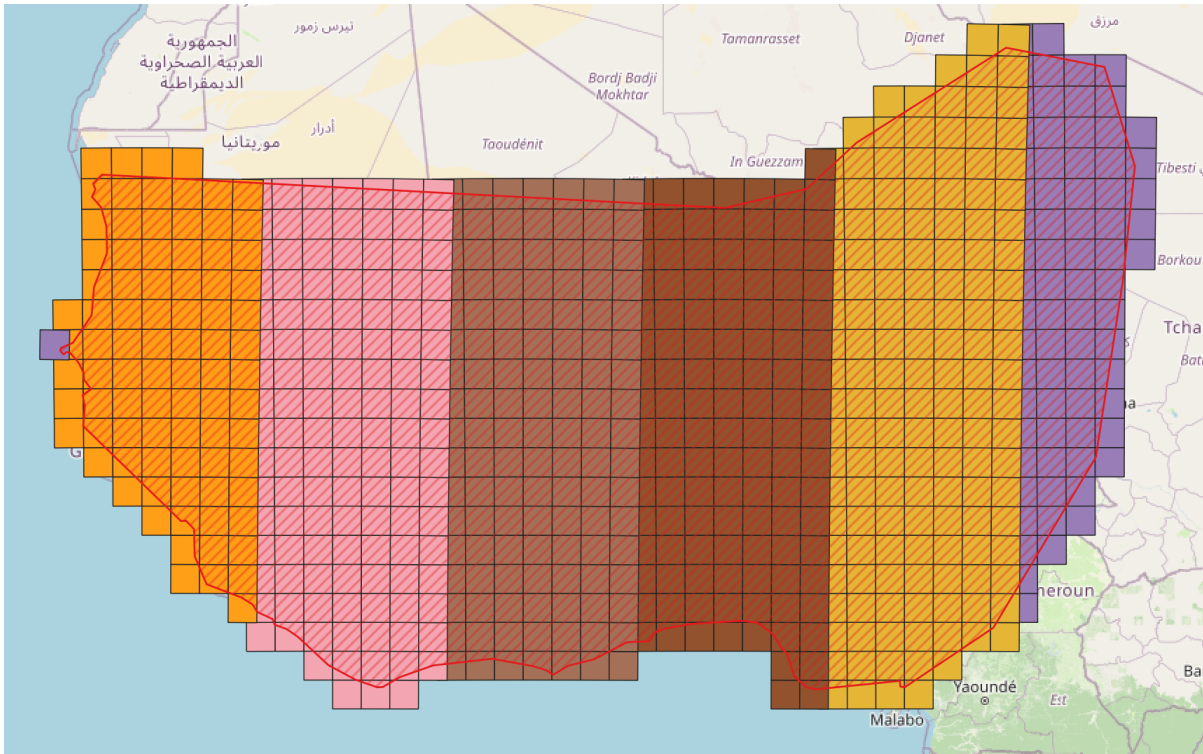
Figure 50: Grid of EOPatches over AoI.

Merging three temporary datacubes with a year worth of data each, we then ended up with EOPatches holding NDWI values for each pixel for temporal interval from 2020-01-01 to 2022-12-31.

3.4.3 Calculating baseline statistics

We have put together an eo-grow pipeline to calculate baseline statistics. The execute method of the main task of the pipeline is shown in Code 3, where NDWI data from all years for each month is joined into mean and standard deviation, and number of valid observations is counted.

After running this pipeline, EOPatches will contain information as seen in Figure 44.

Code 3: EOTask for calculation of baseline statistics.

```python
class AnomalyBaselineTask(EOTask):
    ...

    def execute(self, eopatch) -> EOPatch:
        mean = []
        std = []
        count = []
        for month in range(1, 13):
            month_idx = [idx for idx, ts in
                             enumerate(eopatch.timestamps) if ts.month == month]
            masked_data = np.where(
                eopatch[self.input_feature][month_idx, :] < -1,
                             np.nan,
                             eopatch[self.input_feature][month_idx, :]
            )
            mean.append(np.nanmean(masked_data, axis=0).squeeze())
            std.append(np.nanstd(masked_data, axis=0).squeeze())
            count.append(np.sum(~np.isnan(masked_data), axis=0).squeeze())

        eopatch[self.mean_feature] = np.moveaxis(np.array(mean), 0, -1)
        eopatch[self.std_feature] = np.moveaxis(np.array(std), 0, -1)
        eopatch[self.counts_feature] = np.moveaxis(np.array(count), 0, -1)

        return eopatch
```

3.4.4 Monitoring anomalies

From this point onward, the workflow follows the idea, outlined in the continuous monitoring example, shown in eo-grow-examples repository at https://github.com/sentinel-hub/eo-grow-examples/blob/main/GEM/docs/continuous_monitoring.md:

- for each EOPatch cell update the local catalog of Sentinel-2 (from the last run until "now", or from start in case of the first run, using CatalogPipeline
- download the missing Sentinel-2 NDWI images - the newly available imagery since the last run of the pipeline using IncrementalDownloadPipeline
- calculate anomalies as defined above with the CalculateAnomaliesPipeline. Additionally, as in the next steps the calculated anomalies will be ingested back to Sentinel Hub, the anomalies are reformatted to optimize storage. The main method to compute anomalies is shown in **Code 4**.
- Export anomalies to TIFF files with ExportMapsPipeline
- Ingest resulting TIFF files to a static Sentinel Hub datacube using BYOC API via the IngestByocTilesPipeline

Code 4: Anomaly calculation part of the EOTask.

```python
class AnomaliesTask(EOTask):
    """Calculate anomaly index based on data, its historical mean and standard deviation"""

    ...

    def execute(self, data_eopatch: EOPatch, reference: EOPatch) -> EOPatch:
        anomalies = []
        for tid, ts in enumerate(data_eopatch.timestamps):
            mean = reference[self.mean_feature][..., ts.month - 1]
            std = reference[self.std_feature][..., ts.month - 1]
            counts = reference[self.counts_feature][..., ts.month - 1]

            data = data_eopatch[self.data_feature][tid, ..., 0]

            # anomaly; set to -100 if not enough history or non valid data
            anomaly = np.where((counts > self.min_valid_counts) & (data > -1),
                        (data - mean) / std,
                        -100)

            # optimise storage - map to uint8
            anomaly_uint = np.where(anomaly == -100, 0,
                                        12*(11+np.clip(anomaly,-10,10)))
                        .astype(np.uint8)

            anomalies.append(anomaly_uint)

        data_eopatch[self.output_feature] = np.array(anomalies)[..., np.newaxis]

        return data_eopatch
```

These pipelines can then be run at any cadence (e.g., daily), using a simple CRON job. For instance, the following CRON schedule expression will execute a command every day at noon:

    0 12 * * *

The command would simply be:

    eogrow end2end.json

or, if using a cluster:

    eogrow-ray infrastructure.yaml end2end.json --start --tmux

The end2end.json configuration, holding all pipelines for continuous monitoring in one place, makes use of the chaining pipelines capability of eo-grow:

```
[
  {"**catalog_update": "${config_path}/01_update_catalog.json"},
  {"**incremental_download": "${config_path}/02_incremental_download.json"},
  {"**calculate_anomalies": "${config_path}/03_anomalies.json"},
  {"**export_to_tiff": "${config_path}/04_export_maps.json"},
  {"**ingest_to_SH": "${config_path}/05_ingest_to_sentinelhub.json"}
]
```

Such monitoring has been run for the period 1.1.2023 – 8.31.2023 for the whole Sahel region. Most of this period has been executed in a few runs, but we have tried and tested the script to verify it works when run e.g., daily. The whole concept is agnostic to when the service is "re-run"; it will always pick up from the last time it was executed, and process everything that was new, making it extremely versatile. Such incremental updating is also resilient to data delivery interrupts, e.g., to delays in Sentinel-2 data delivery from ESA.

3.5 Results

The main objective of this exercise was to show the capabilities of the GEM tools for continuous monitoring. The results of this particular exercise are not as outstanding as one would hope since the focus was on the processing steps. As we can see from a mosaicks of anomalies for April and May in Figure 51 and Figure 52, large trends of dry vs wet areas can still be seen. Comparing our results to similar but detailed research, at such scales they are often done on significantly lower resolution (1km grid), but also use significantly longer periods to construct historic baseline statistics. Indeed, the latter seems to be a big issue over the cloudy equatorial and monsoon regions, where the number of valid observations per month interval was below a defined minimal value min_valid_counts (we used 8 in our runs). Interestingly enough, in areas where Sentinel-2 orbits overlap, we often got above this limit, hence the narrow southward pointing strips in Figure 51 and Figure 52.
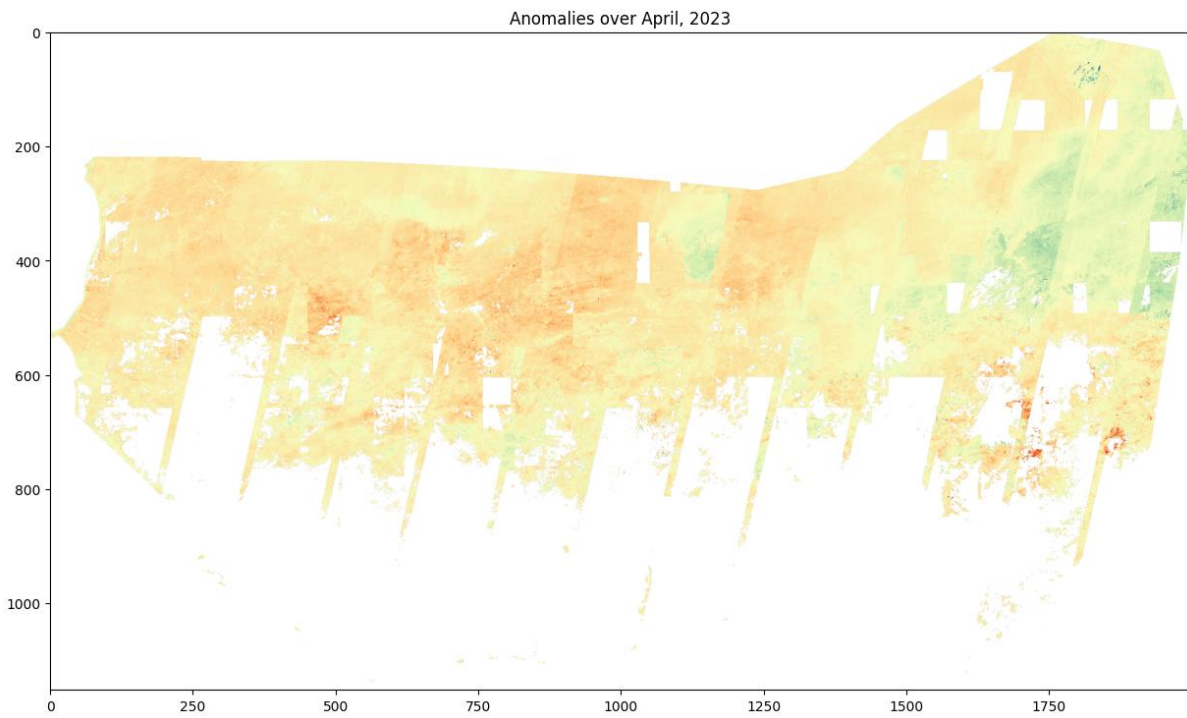
Figure 51: NDWI anomalies from observations in April 2023.
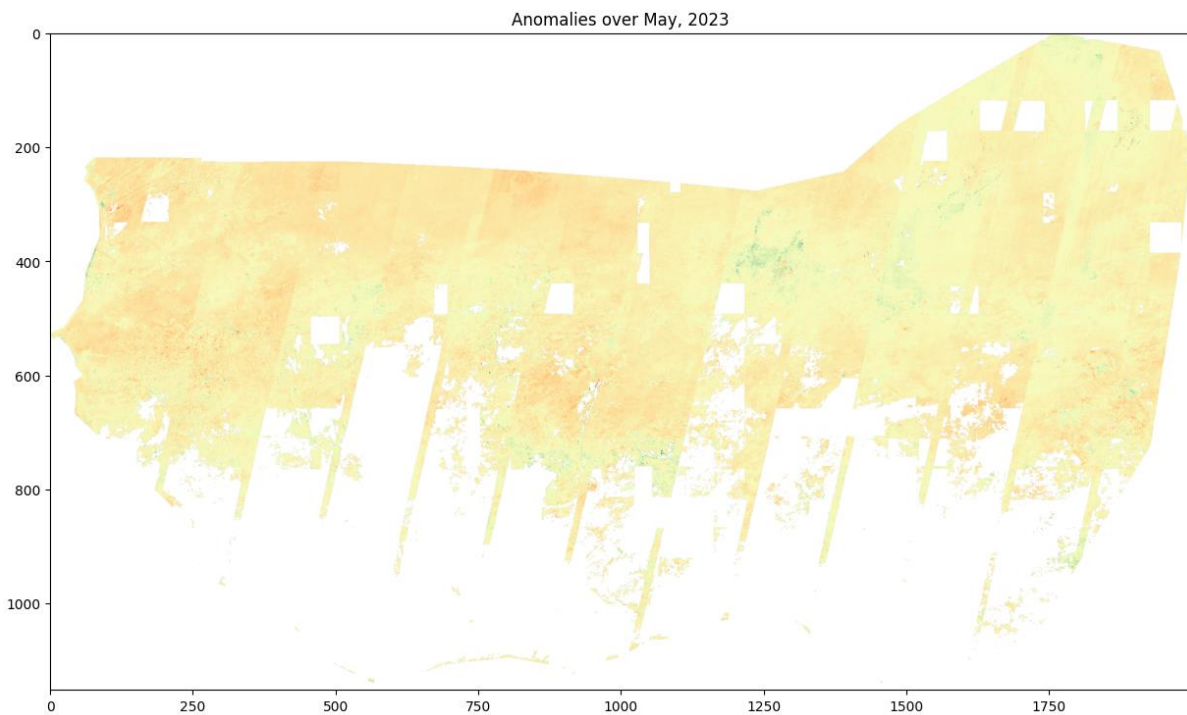
Figure 52: NDWI anomalies from observations in May 2023.

Additionally, when looking at results at full resolution, it becomes transparent that cloud shadows pose an issue – the NDWI calculation is returning significantly higher NDWI values over areas in cloud shadows, shown in Figure 53.
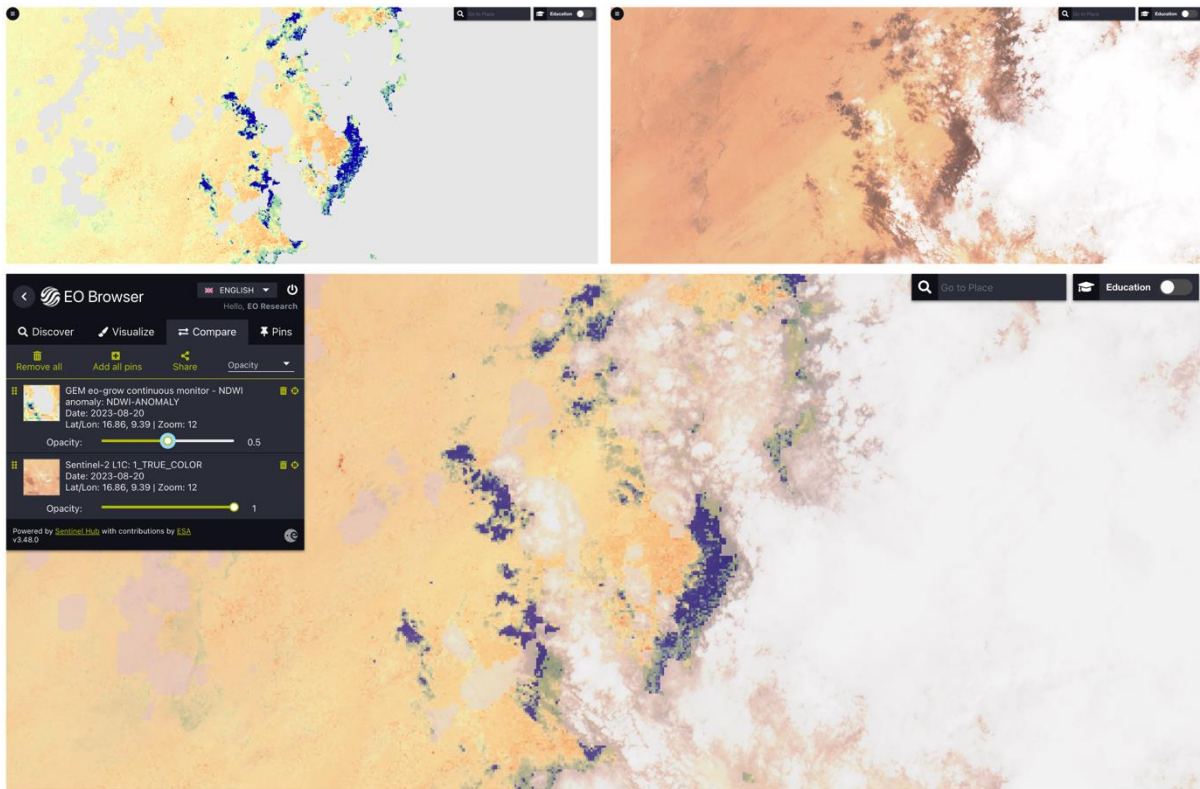
Figure 53: Effects of cloud shadows to anomaly detection. Top left: anomalies, top right: true color Sentinel-2 L1C image, bottom: anomaly overlaid over true color.

Additional benefit of the GEM framework becomes clear when the results of the analysis are incorporated back to Sentinel Hub. At that point, the full API of Sentinel Hub can be used, and results easily visualized, compared, re-used, etc. Figure 53 already showcases such integration: using EO-Browser, overlaying results from anomaly detection with Sentinel-2 true color imagery is a matter of seconds.

The results, calculated NDWI anomalies for each Sentinel-2 observation over AoI, are available for access through the GEM framework. The appropriate data cube / data collection to be used is:

ndwi_anomaly = DataCollection.define_byoc("37d1fa19-9aaf-49a4-98f9-20daf313040f"),

and a notebook illustrating how Figure 46 can be created is available on eo-learn-examples within GEM-data and shown in Figure 54.

Notebook in eo-learn-examples[21] contains examples for retrieval of whole time-series of anomalies, as well as a mosaick shown in in Figure 51.

---

[21] https://github.com/sentinel-hub/eo-learn-examples/blob/main/GEM-data/gem-NDWI-anomalies.ipynb

```python
from sentinelhub import SentinelHubRequest, BBox, CRS, DataCollection, MimeType
```

```python
ndwi_anomaly_data_cube = DataCollection.define_byoc("37d1fa19-9aaf-49a4-98f9-20daf313040f")
```

```python
viz_anomaly_evalscript = """ •••
```

```python
request = SentinelHubRequest(
    evalscript = viz_anomaly_evalscript,
    input_data=[SentinelHubRequest.input_data(
        data_collection=ndwi_anomaly_data_cube,
        time_interval=["2023-06-01","2023-06-30"]
    )],
    responses=[SentinelHubRequest.output_response("default", MimeType.PNG)],
    bbox=BBox((-17.569541605,4.231063941,16.251236737,23.712313469), CRS.WGS84),
    size=(2000,1152)
)
```

```python
fig, ax = plt.subplots(figsize=(10,8))
ax.imshow(image)
ax.set_title("Anomalies over April, 2023")
```

```
Text(0.5, 1.0, 'Anomalies over April, 2023')
```
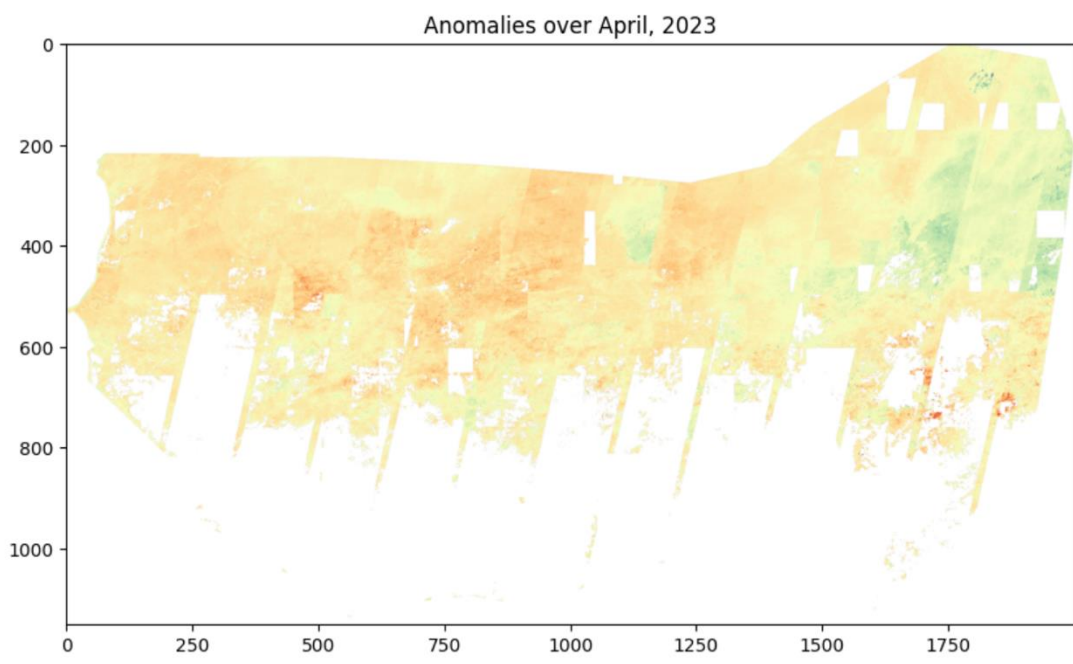

Figure 54: Retrieving anomaly results using Python in Jupyter notebook.

3.6 Summary

The main objective of the continuous monitoring service was to see how GEM framework can be employed for continuous monitoring. We have focused on monitoring water over Sahel region, and first built a historical baseline, and then run the algorithm to calculate anomalies in NDWI for each new Sentinel-2 observation continuously. With the example shown in this section we believe the objective was reached, despite the results, the NDWI anomalies, not being outstanding.

The approach using eo-grow is also very nimble: with simple tweaks to the pipelines, or even just configuration files, other models could be used, and results monitored.

Some examples:

- instead of simple baseline statistics, the parameters for modeling of seasonality of vegetation index could be calculated from historical data, and then each new observation could be checked for large deviances from predicted values. This way, eo-grow based BFAST[22] monitor could be employed.
- NDWI based on near infrared and SWIR band, as proposed by Gao[23] could be used in the same approach to monitor leaf (vegetation) water content at canopy level.
- Vegetation index could be used to find deforestation.
- Fancy deep-learning model could be used to predict something (e.g., deforestation), and ongoing monitoring part could compare the last prediction to previous, looking for changes.

Additionally, other datasets could be used; as the service relies on eo-grow, tweaking the configuration for the download pipeline would be sufficient to switch from Sentinel-2 to Sentinel-1 (e.g., to remove issues with clouds) or use Harmonized LandSat Sentinel dataset (to obtain longer historical baseline).

---

[22] Breaks For Additive Season and Trend, http://bfast.r-forge.r-project.org
[23] Gao, B.-C. 1996. NDWI - A normalized difference water index for remote sensing of vegetation liquid water from space. Remote Sensing of Environment 58: 257-266.

4 Demonstration Applications

The following section shows the dedicated clients developed to visualize, compare, and validate results for each GEM use case.

4.1 LC-CMS Application

To demonstrate the results of this use case over the pilot regions, two simple web-based Map Viewers have been created: The first one is meant for internal use as it covers additional context and layers which are private to TomTom. The second is shared with the GEM Consortium partners on the GEM website, to visualize the pilot results for the LC-CMS use case. The app is currently available at:

http://globalearthmonitor.eu/sites/default/files/LC_CMS/index.html
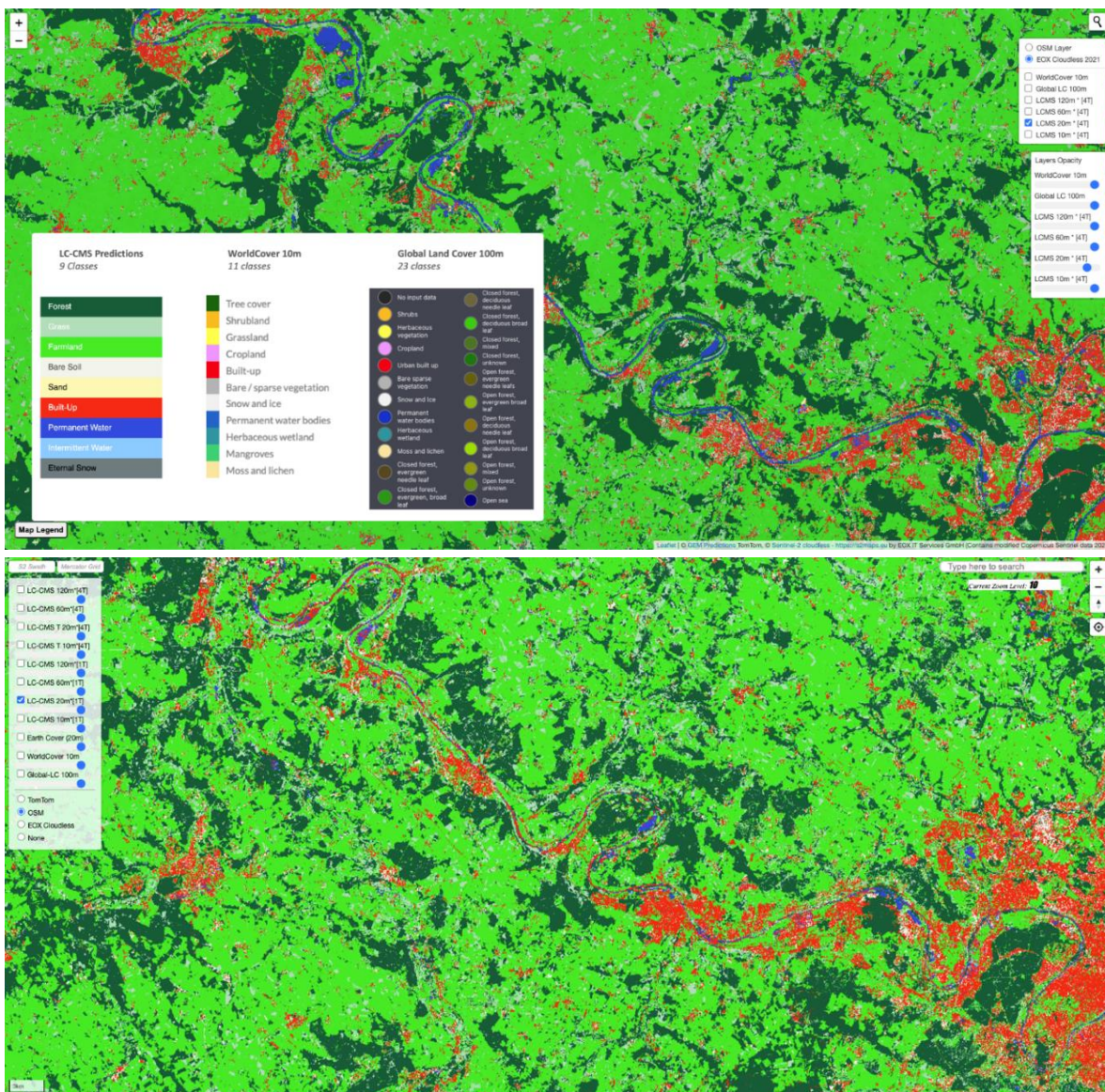

Figure 55. The public viewer (top) and internal viewer (bottom).

Both viewers are continually evolving, with new layers and functionalities added. Figure 55 shows the internal and the public viewers. While the both viewers share same functionalities, their main difference is the underlying libraries e.g., the internal viewer uses MapLibre GL JS, while the public viewer uses open-source Leaflet JS.

As seen in the Figure 55, there are multiple layers in the viewer which can be turned on and off using the toggle button for each layer. The slider controls can be used to change the opacity of each layer which makes it easier to compare results of different layers. In addition to that, the viewer also includes a Map Legend which shows the colours used for different LC classes in each layer. Internal viewer also includes the TomTom's Earth Cover reference layer, which is not available on the public viewer due to licensing.

The viewer includes the layers shown in Table 4.1.

Table 4.1 Different layers used in LC-CMS Viewer

Layer Name	Layer Type	Description
LC-CMS Prediction results at 120m, 60m & 20m	Prediction Results Layer	Results for predictions performed at 120m, 60m & 20m.
LCMS Pilot 120m 2019, LCMS Pilot 120m 2020, LCMS Pilot 120m 2021	Prediction Results Layers	Prediction results for the Pilot region performed by LC-CMS pipeline using S2 at 120m for 2019, 2020 and 2021.
LCMS Change 2019-2020, LCMS Change 2020-2021, LCMS Change 2019-2021	Change Detection Results layer	Change detection by comparing predictions performed by LC-CMS pipeline using S2 at 120m for 2019, 2020 and 2021, over the years.
TomTom Base Map	Reference Layer	TomTom base map raster layer.
OSM Layer	Reference Layer	OSM (Open Steet Map) raster base map layer.
Cloudless Satellite Imagery	Reference Layer	Pre-processed Cloudless Viewing Ready (RGB) global product created from Sentinel-2 mission for years 2019, 2020, 2021.
World Cover 10m	Validation Layer	The European Space Agency (ESA)'s World Cover maps which is global land cover map with 11 different land cover classes produced at 10m resolution based on combination of both Sentinel-1 and Sentinel-2 data.
Global-LC 100m	Validation Layer	100m LC product delivered by Copernicus for years 2015-2019.

4.2 Map Making Application
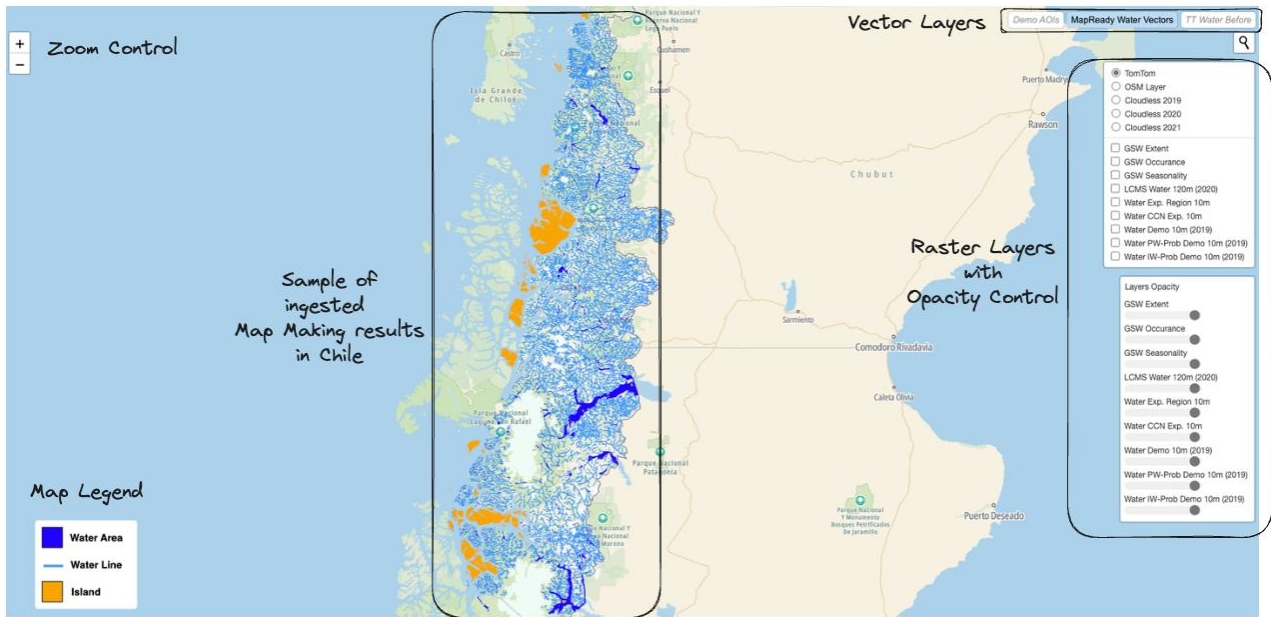
Figure 56 Map Making results viewer.

Like LC-CMS viewer, we created a dedicated viewer for visualising the results for Map Making use case. The viewer is available at http://globalearthmonitor.eu/sites/default/files/water/index.html.

Table 4.2 shows the list of the layers used for demonstrating the results of Map Making use case. Figure 56 shows the viewer with following features:

- Vector layers are displayed on the top right row of the viewers and can be displayed using the toggle buttons.

- Raster layers can be displayed selecting the checkbox with the layer name.

- Reference layers can also be changed using the radio buttons with layer name. Opacity of the selected layer can be changed with slider associated with the layer name.

Table 4.2 List of layers for demonstrating the results of Map Making use case.

Layer Name	Layer Type	Description
WaPr 60, WaPr 20m, WaPr 10m	*Prediction Results Layers*	Prediction results for the Myanmar experiment region performed by Water Predictor pipeline using S2 at 120m, 20m and 10m resolution for the year 2020
MapReady Water Vectors	*Vector Layer Results*	This layer shows a sample of map-ready vectorized results generated using the Map Making pipeline for a small region of Chile. Data for whole of Chile has been ingested into TT map and this is a subset of the ingested data.

TT Water Before	*Reference Layer*	Reference Layer which shows the situation of TT map in the demo region of Chile. This data has now been replaced in TT map with new ingested data from Map Making pipeline.
WaPr 10m FED	*Prediction* Results Layers	Prediction results for the Myanmar and Chile demonstration region performed by Water Predictor pipeline feature engineering on 10m resolution for the year 2020 (Myanmar) and 2021 (Chile)
WaPr 10m CNN	*Prediction* Results Layers	Prediction results for the Myanmar and Chile demonstration region performed by Gap Filling CNN Segmentation model defined in D5.4 in section 4.3 on 10m resolution for the year 2020 (Myanmar) and 2021 (Chile)
WaPr Change 2019-2022	*Change Detection* Results layer	Change detection by comparing predictions performed by LC-CMS pipeline using S2 at 120m for 2019 and 2022 for drying lakes in Iran and Iraq.
GSW*	*Validation* Layer	GSW Layers form the open sourced Global Surface Water Explorer (global-surface-water.appspot.com)
TomTom Base Map	*Reference* Layer	TomTom base map raster layer.
OSM Layer	*Reference* Layer	OSM (Open Steet Map) raster base map layer.
Cloudless Satellite Imagery	*Reference* Layer	Pre-processed Cloudless Viewing Ready (RGB) global product created from Sentinel-2 mission for years 2019, 2020, 2021.

4.3 Built-Up Application
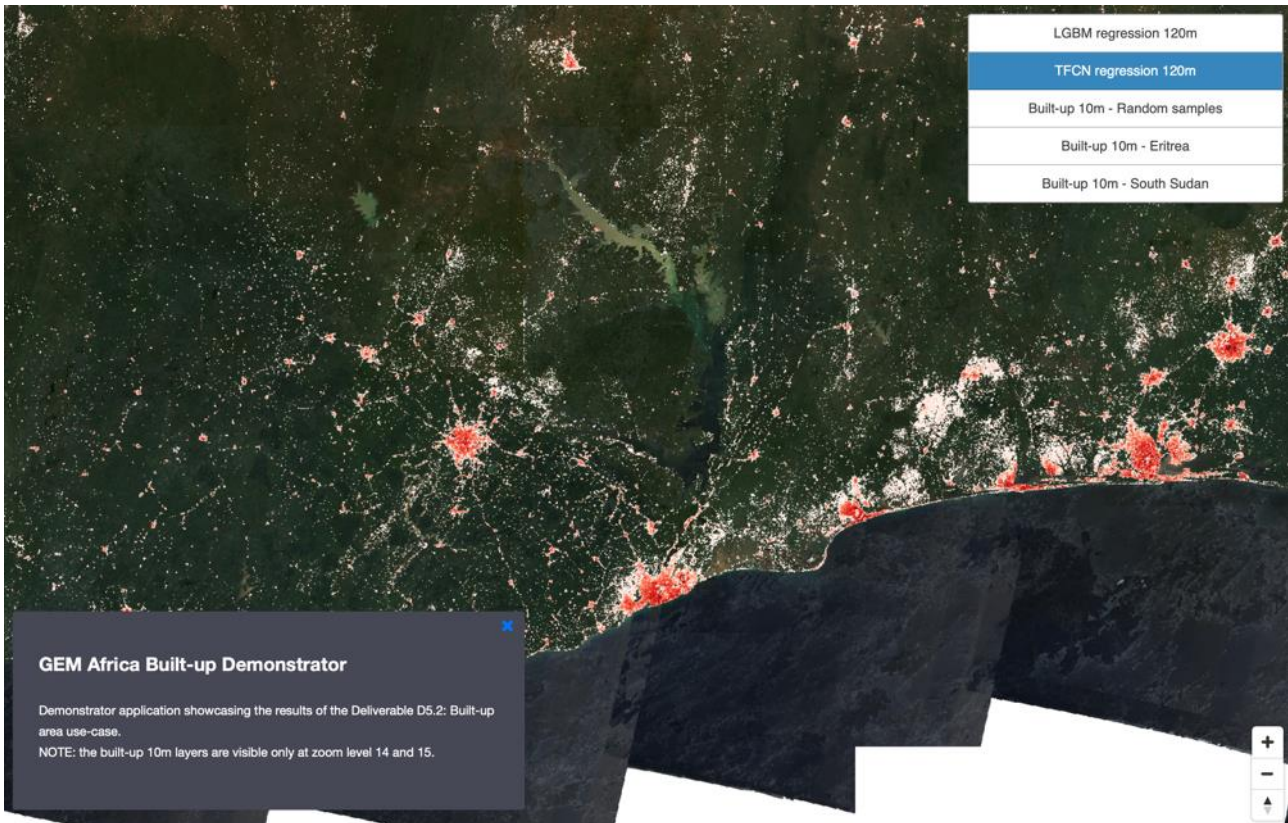
Figure 57: Screenshot of the Built-up demonstrator app.

Demonstrator of the Built-up use case results obtained over Africa is currently available at:

https://www.globalearthmonitor.eu/sites/default/files/D5.2_demo.html

The demo presents results of two research modalities:

Data and results obtained at 120 m resolution, showing Sentinel-2 120 m global mosaic as backdrop, and results of two best performing ML models as different layers for the user to visualize:

- LightGBM regression model, presented in section 2.3.5.1

- Temporal Fully Connected Network with regression output, presented in section 2.3.5.2.

Results obtained at 10 m resolution is shown for:

- areas from different biomes across whole Africa

- South Sudan

- Eritrea

The results at 10 m resolution use Sentinel-2 Level2A data at 10 m as backdrop, and the polygons are results of predicting bounding boxes of buildings using Sentinel-2 data, described in section 2.3.6.

All the raster data in the demonstrator is being retrieved from Sentinel Hub services, where the results are ingested as BYOC (Bring Your Own Cloud-optimised GeoTIFF) data.

4.4 Crop Identification Application

The demonstrator for the Crop Identification use case is a simple web application[24] that allows users to inspect the results of crop identification experiments. It is configured to show the most accurate results we obtained, i.e. by training with Slovenia 2019 crop data.

The demonstrator allows users to zoom and pan a map to a region of interest. They can then click on individual fields to inspect classification results (Figure 58).
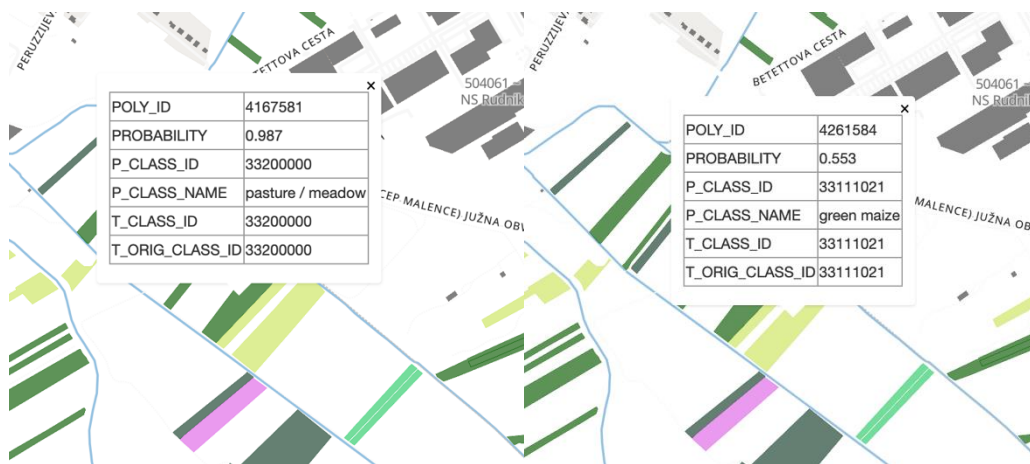

Figure 58: Crop identification results map. Crop types are colour coded; clicking on a crop shows crop identification results including the probability of the crop being identified correctly. Pasture/meadow can be identified with high probabilities (left); identification of other crops may be not as reliable (right).

For each field, the application provides the following information:

- Polygon id
- Probability with which the crop was predicted
- Predicted crop id
- Predicted crop name
- Target crop id (from crop database)

This demonstrator is freely accessible for review and will be hosted on the GEM website.

4.5 Temperature Downscaling

meteoblue has matured the demonstrator for temperature downscaling for built-up areas to TRL 9 and displays live heat maps in of selected cities[25] in 10 m resolution for free on the meteoblue web site. Ljubljana is shown as an example. The colour scale is chosen to amplify differences – at different times of day a particular hue may refer to different temperatures.

---

[24] https://www.globalearthmonitor.eu/sites/default/files/crop_identification/index.html#14.62/45.9582/14.35993
[25] https://www.meteoblue.com/cityclimate

Figure 59: Cities for which meteoblue.com shows free 10 m hyper-resolution temperature maps[25]

Figure 60: Hyper-resolution (10 m) temperature map of Budapest on September 6, 2023, at 04:00h . The temperature spread at that time was 14 °C (blueish) to 19 °C (dark brown).

4.6 Compute climate variables on grids

meteoblue operates a self-service portal[26] for simulated historic weather data access that also supports server side computation. This allows users to obtain climate data as derived variables directly in the portal. For GEM, meteoblue created and documented example Python scripts[27].

This demonstrator is freely accessible.

---

[26] link to self service portal

[27] https://github.com/sentinel-hub/eo-learn-examples/tree/main/climatological_days

4.7 Measurements interface

The user interface for the meteoblue Measurements API has been reported in D2.4 and is repeated here for completeness. For easy access to measurement data meteoblue provides a web interface[28] (Figure 61) that allows users to easily access measurements and lets them find and select stations by selecting criteria in the web interface. Users can:

- have an overview over available stations;

- select stations within a specific geographic area represented by polygons;

- select stations with a minimum hourly coverage of a specific weather variable during a specified time interval;

- select download formats and download the data.

Figure 61: meteoblue measurements interface: overview (top), political area selection (bottom left), and provider selection (bottom right) screens.

---

[28] https://measurements.meteoblue.io/measurements-ui/#/map?providers=synop,metar

This user interface does not yet support selective data access based on user credentials. It is therefore only suitable for internal use by meteoblue employees or external use by customers with a special contract.

4.8 Conflict Pre-Warning Application

4.8.1 Conflict dashboard

As described in D5.5, different datasets have been integrated in a dashboard for visualization, aggregation and analytics. In particular, conflict data has been consumed through the API services provided by the ACLED project and ingested into an ElasticSearch cluster. In this way, GEOINT analysts can perform queries, faceting by different topics and aggregations on an ad-hoc basis, accordingly to the specific GEOINT analyst needs. On top of the Elasticsearch cluster, a Kibana dashboard has been created to facilitate the use and provide self-analytics capabilities. This data is an essential part for the analyses in the CPW use case, as it has been thoroughly used for the validation of the products and services generated through the exploitation of EO, meteorological and collateral datasets.

The dashboard is interactive, meaning that it is able to dynamically adapt to the queries posed by the analyst and showcase the different facets and aggregations allowing to understand different situations, from wide temporal extensions and global scenarios to very localized issues in time and geospatial location.
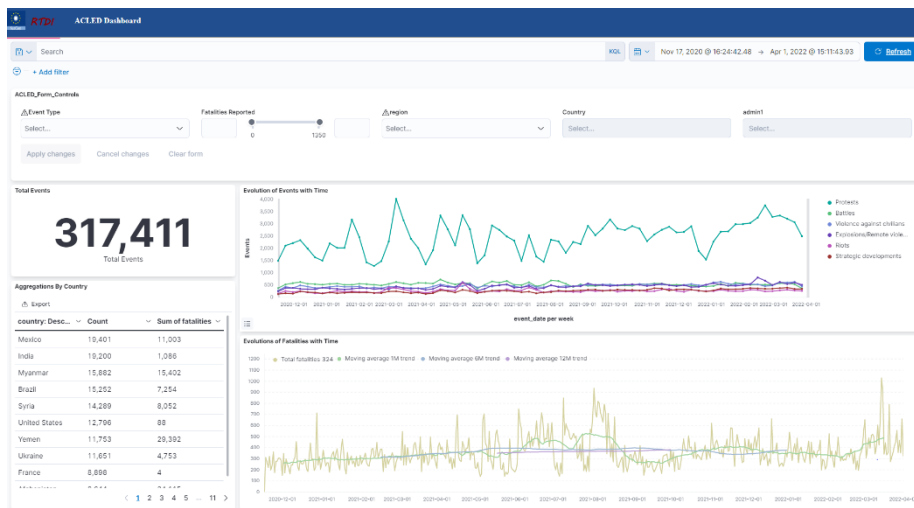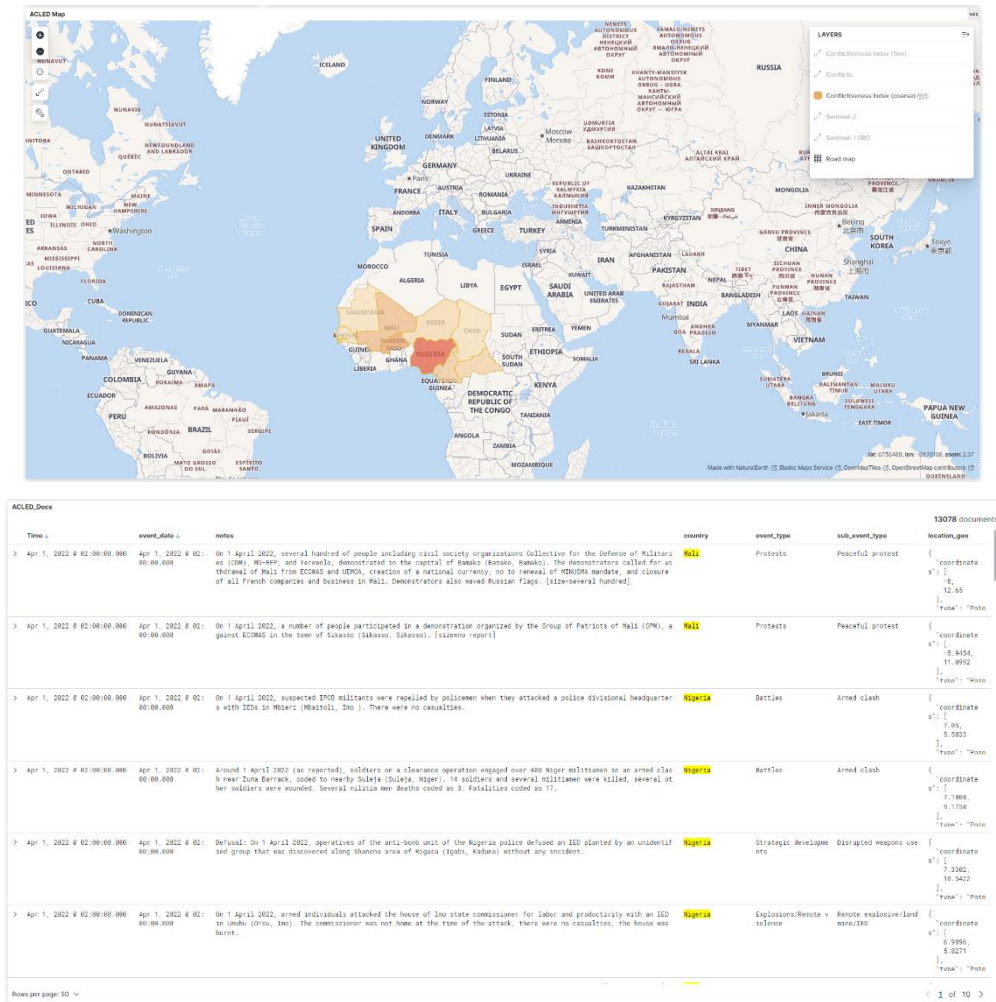
Figure 62: CPW Dashboard

The system allows filtering data by different constraints, including country, region, event types or subtypes, number of fatalities between specific range, and even free text search on the description of the event.

The interface automatically presents the timeline and facts such as the top countries involved or top actors involved because of certain query.

It also provides a map view that seamlessly allows drilling down into details providing an aggregated value taking into consideration the level of zoom. At high levels of zoom, the data is aggregated at the country scale. Once the user zooms in, the data is disaggregated into a cluster grid. On further zoom, the individual events are showcased, with the colour representing the type of event and the size representing the number (or estimate) of fatalities related to that event.

Different layers have been included to enable the analysis of conflicts with other datasets. The data is also exportable to geospatial formats (e.g., geojson) for integration in traditional GIS software (QGIS, ArcGIS).
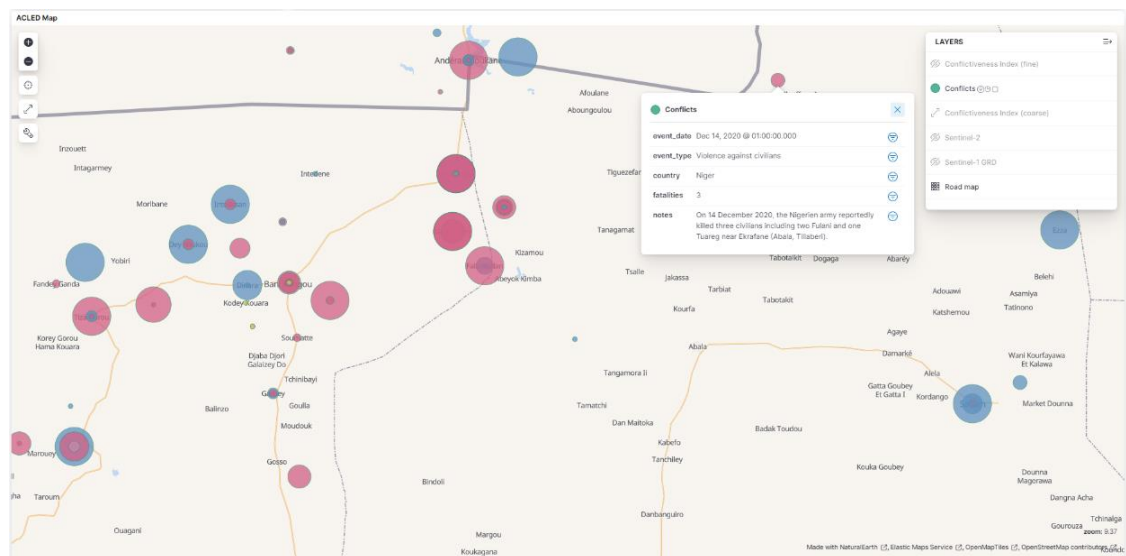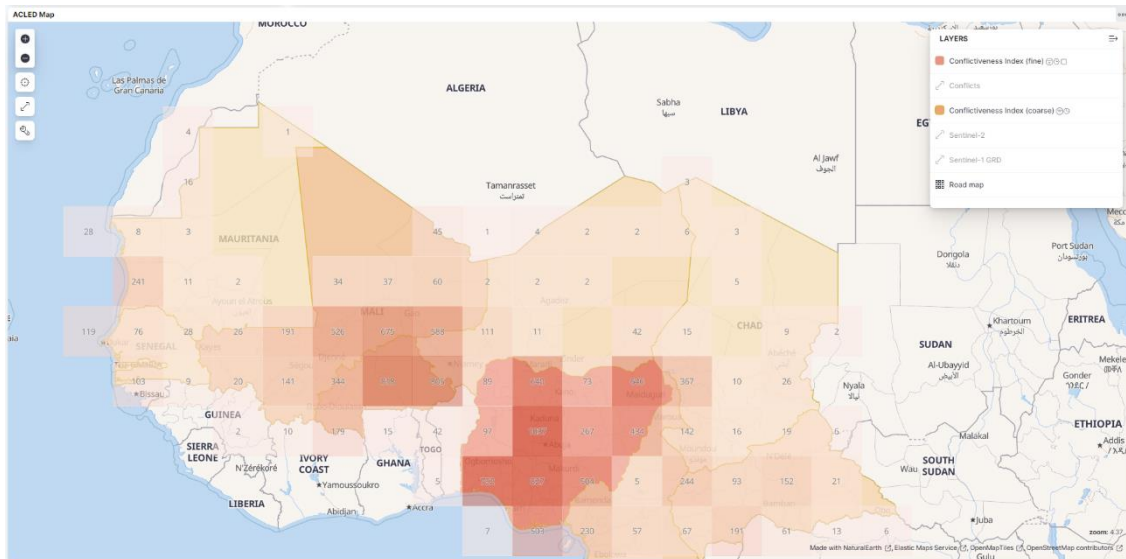
Figure 63: Cross-scale analysis for conflict data

Additional derived layers are also available for visualization in the platform, such as water body transitions (See Figure 64) and Land Cover maps/changes (see Figure 65).
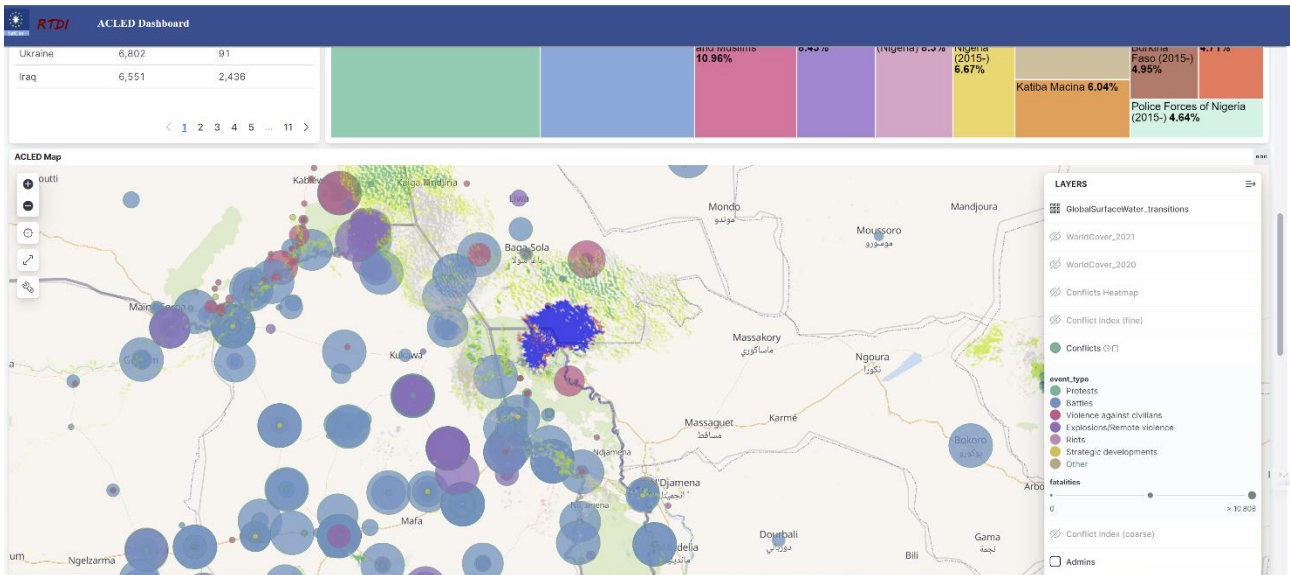
Figure 64: Visualization of water body transitions and conflict events around Lake Chad Basin for a specific period of time
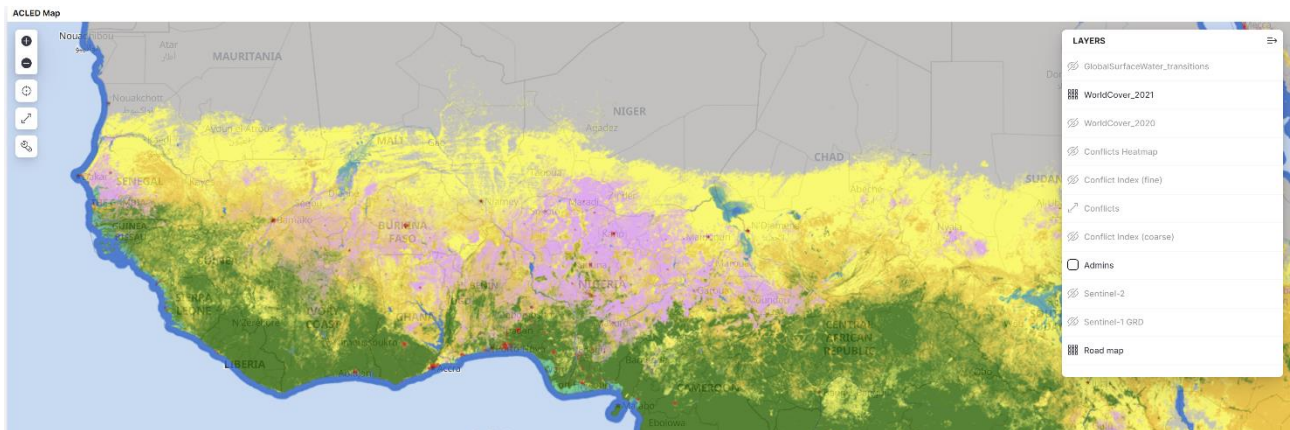
Figure 65: Land Cover Map visualized in the prototype interface

A self-service analytics module is also available, allowing users to interact with the data in a way that they can generate new views that can then be used on their own or further integrated in the dashboard. By simple drag & drop (see Figure 66) new visualizations can be created. Panel (1) allows for filtering/slicing the data. Panel (2) allows users to define the metrics (accumulated – sum - number of fatalities in the example) and facets (by time in the example). Panel (3) allows for visualization of the data, in the form of a line chart in the example.
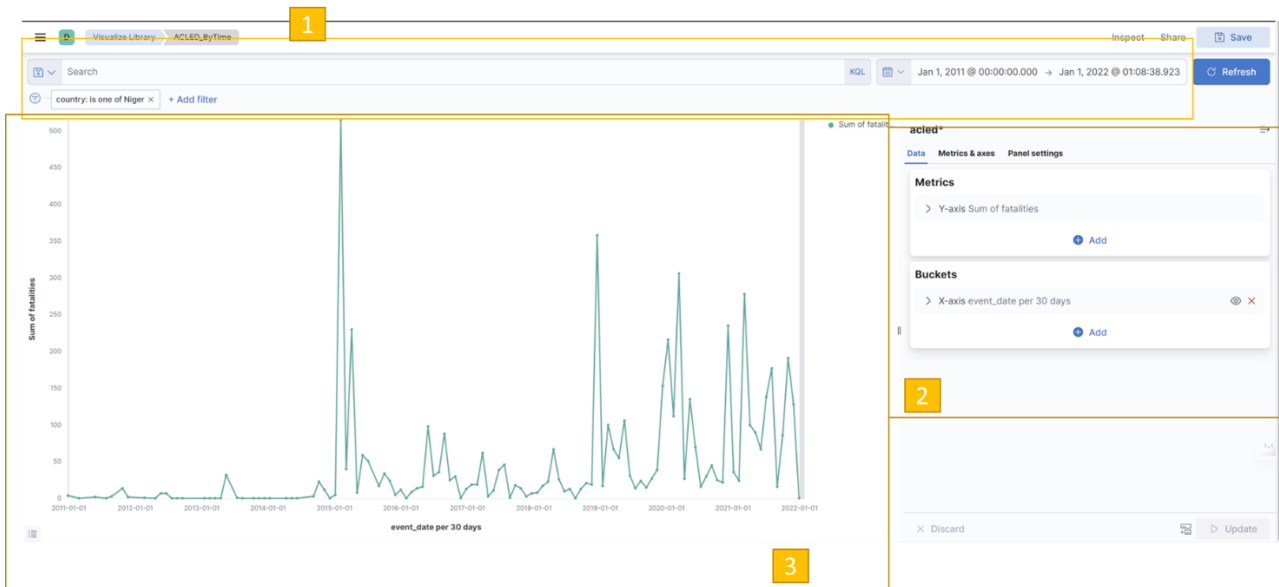
Figure 66: Self-service analytics interface

4.8.2 Jupyter notebooks

For expert users, the data is also offered to be consumed using standard programming interfaces. SatCen has also deployed a Jupyter Lab instance so that users can access the data programmatically and create their own visualizations. An example can be found in Figure 67 for visualizing the evolution of the average NDVI over a specific area, using Sentinel-2 and GEM capabilities.
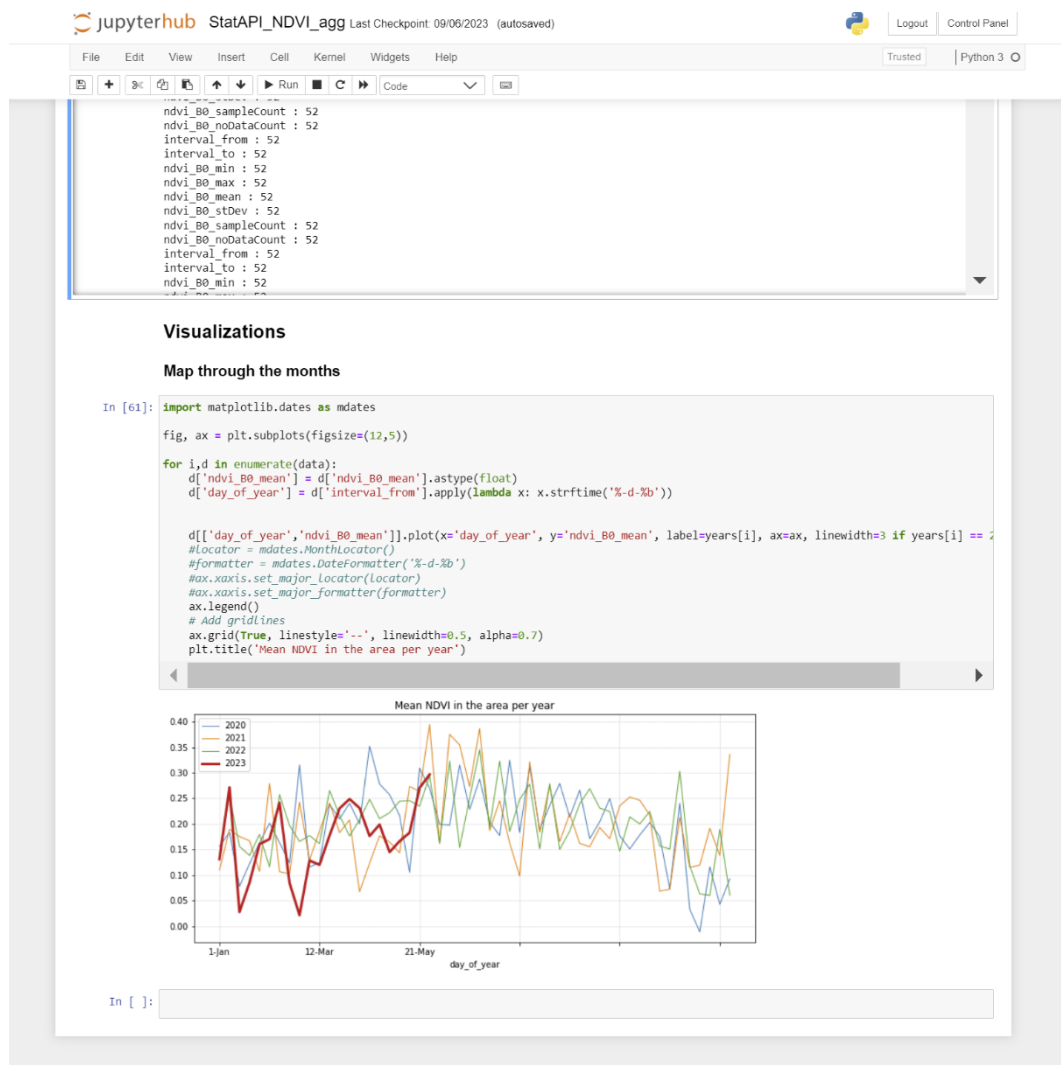
Figure 67: Example of custom code to visualize the evolution of the average NDVI from Sentinel-2 in a specific area

4.9 City Heat Map Application

Cities are known to be "hotter" than the surrounding countryside, and this effect is increased by climate change. It affects more than 50% of the human population - worldwide.

meteoblue demonstrates hyper-resolution (10m) temperature fields as city heatmaps on its website, e.g., at https://www.meteoblue.com/en/products/cityclimate/heatmaps/basel. A screen shot is shown in Figure 68. A time slider allows to inspect different times of day. Notice that high temperature spreads are typically observed in the late morning (caused by differential heating of different surfaces by the sun) and mid-evening (due different cooling of different surfaces once the sunlight is gone).

Figure 68: Real-time city heat map for Basel. Its two major railway stations are easily identified in the centre image. The colour scale is optimised for the temperature spread. Users have to check the colour bar to see the actual temperature values.

This new meteoblue website feature shows the current temperatures in 10x10 meters resolution including a 3-day history for over 70 selected cities and their surrounding areas. The city climate overview map[29] shows all cities for which heat maps are available. Click one of the orange dots to view the corresponding city's hyper-local temperature weather map. The underlying data show local temperature patterns within a city such as heat islands or cool spots. Temperature differences of up to 10°C within only a few meters are possible during extremely hot days, calm weather, and nights.

Besides helping citizens in their daily life during heat or cold waves to find the most comfortable spots, our heat maps also create awareness for the well-explored urban heat island effect. The heat maps can help city administrations to identify and localise heat islands and subsequently plan climate-friendly adaption measures.

---

[29] https://www.meteoblue.com/en/products/cityclimate/index

5 Public Collections

Majority of datasets that have are available (or were made available) to the GEM consortium or publicly are presented in D2.2 – Data Management Plan. The datasets, representing results from the demonstration activities, are presented here to be concise. For many of them, the examples notebooks at https://github.com/sentinel-hub/eo-learn-examples/blob/main/GEM-data/introduction.ipynb showcase how the data can be used within GEM framework.

Table 5.1 lists the Sentinel Hub collections with results from the Built-up use-case.

Table 5.1 Sentinel Hub collections for the demonstration and experiments from Built-up use-case

Layer	Region	Resolution	SH Collection Id	Raster Band(s)
Sentinel-2 L2A 120m Mosaic[30]	worldwide	120m	484d8dbb-9e3e-41f2-b96b-35189d3ae37f	All Sentinel-2 bands
Rasterised "Open buildings" dataset by Google	Africa	120m	6701abce-6aff-448a-b149-ab285620f135	google
LightGBM regression built-up model	Africa	120m	6701abce-6aff-448a-b149-ab285620f135	lgbm_rmse
LightGBM categorical built-up model	Africa	120m	6701abce-6aff-448a-b149-ab285620f135	lgbm_cat
TFCN regression and classification built-up models	Africa	120m	6701abce-6aff-448a-b149-ab285620f135	tfcn_r, tfcn_rc, tfcn_2out_r

Table 4.1 and 4.2 lists all the Sentinel Hub collections which contain the results and predictions for the LC-CMS and Map Making use case respectively.

Table 5.2 Sentinel Hub collections for the demonstration and experiments predictions for LC-CMS pipeline

Layer	Region	Resolution	SH Collection Id	Raster Band
LC-CMS Predictions	France AOI	120m	59b4de97-ad51-47b0-80f6-c6c47fae1ca7	B1
LC-CMS Predictions	France AOI	60m	84f1e9db-4abe-4ddf-8d00-17b2783531c4	B1
LC-CMS Predictions	France AOI	20m	5c21b111-07d2-43a3-95ad-46381c6e199a	B1
LC-CMS Predictions 2019	Africa AOI	120m	d90fec4d-b21b-4780-b0c5-20dc2d1017d4	Y2019

---

[30] https://collections.sentinel-hub.com/sentinel-s2-l2a-mosaic-120/

LC-CMS Predictions 2020	Africa AOI	120m	d90fec4d-b21b-4780-b0c5-20dc2d1017d4	Y2020
LC-CMS Predictions 2021	Africa AOI	120m	d90fec4d-b21b-4780-b0c5-20dc2d1017d4	Y2021

Table 5.3 Sentinel Hub Collections for demonstration and experiments predictions for Map Making use case

Layer	Region	Resolution	SH Collection Id	Raster Band
Water Demo AOIs Prediction 2019	All AOIs mentioned in table 2.4	10m	f234133c-d198-4569-a14f-612d692f2030	PRED
Probabilities for PW 2019	Chile AOI	10m	6da384a4-1ea1-40a6-b855-0b4ad954671f	PW
Probabilities for IW 2019	Chile AOI	10m	6da384a4-1ea1-40a6-b855-0b4ad954671f	IW
Water CNN Gap Filling Predictions	Chile AOI	10m	b31d5c95-ddb6-440f-b732-5827f5f64fb7	PRED
Water Experiments Predictions 2020	Myanmar AOI	10m	742682e8-ab52-46b6-bfa8-6fc068cb60cf	PRED

Table 5.4 shows the collection holding the NDWI anomalies data, results of the continuous monitoring service, described in Section 3.

Table 5.4 Sentinel Hub Collections holding results from continuous monitoring service

Layer	Region	Resolution	SH Collection Id	Raster Band
NDWI anomalies	Sahel AOI, shown in Figure 49	120m	37d1fa19-9aaf-49a4-98f9-20daf313040f	anomaly

Vectorized Results of Map Making for Demo Regions

The vectorized map-ready results generated using the Map Making pipeline for all the demo regions are available for download via GEM web page.

There are results for 5 regions mentioned in table 2.4. As mentioned earlier, the results of South Korea cannot be shared because of contractual obligations of TomTom:

- https://globalearthmonitor.eu/sites/default/files/map_making_data/Albania_AOI.gpkg

- https://globalearthmonitor.eu/sites/default/files/map_making_data/Chile_AOI.gpkg

- https://globalearthmonitor.eu/sites/default/files/map_making_data/Taiwan_AOI.gpkg

- https://globalearthmonitor.eu/sites/default/files/map_making_data/Turkey_AOI.gpkg

- https://globalearthmonitor.eu/sites/default/files/map_making_data/Uruguay_AOI.gpkg

Each GPKG file contains the vector results of an AOI and contains three layers, as shown for Albania AOI in Figure 69:

1. WAs: Water Areas

2. WLs: Water Lines

3. Islands: Islands
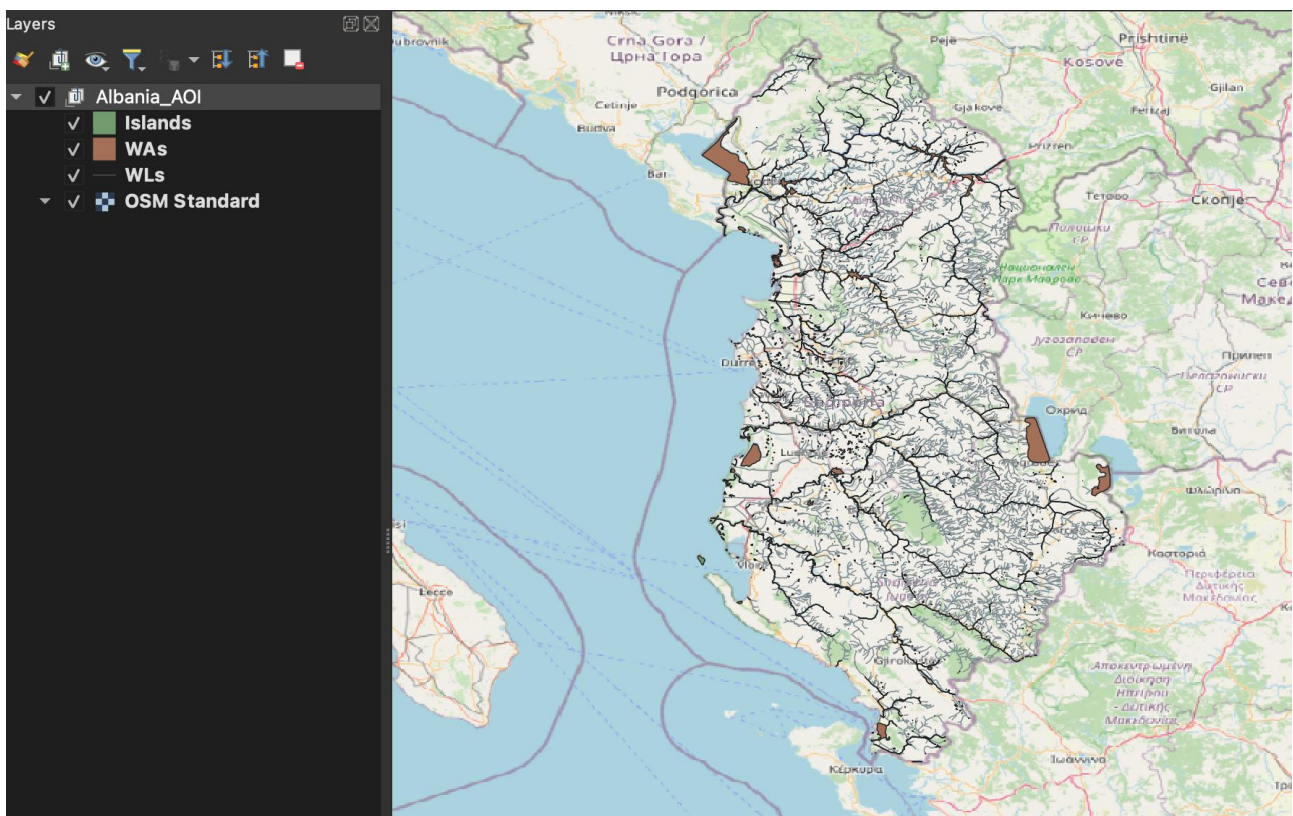
Figure 69: Vector results for Albania AOI shown in QGIS.

A sample (within Chile) of the shared results is shown on the Map Making map viewer application (section 3.2) as a vector layer which can be visualized by clicking on 'MapReady Water Vectors' button on the viewer. A reference layer for the same region is also provided which contains the situation of water bodies before they were updated by GEM Map Making pipeline.

The example result of crop identification is available for inspection via a simple web application on the GEM website (Section 4.4). Hyper-resolution temperature maps for selected cities are available on the meteoblue website (Section 4.5).

6 Conclusion

As a part of LC-CMS use case we built an E2E LC-CMS pipeline using the GEM processing framework, and existing know-how. We worked in close collaboration with Sinergise who developed the GEM processing framework, on which LC-CMS is based. TUM also provided feedback for ML solutions implementation.

The Sahara region of Africa was chosen for the demonstration. To demonstrate Continuous Monitoring of the LC-CMS pipeline, continuous predictions are generated over 3 years from 2019 to 2021 using Sentinel-2 120m resolution data aggregated every 2 months. The public demonstration app described in D5.3 section 2 is used to visualize demonstration region results.

As a part of LC-CMS we also looked at various approaches for Change Detection (CD) and demonstrated the result of change detection using prediction delta. For CD using prediction delta, the CD pipeline was run, and results are shown for the same Africa AOI continuously for years 2019 through 2021 at 120m resolution. We found that CD using prediction delta worked very well for most LC classes in geographically diverse regions. Therefore, we later adapted and created a pipeline of its own as part of the Map Making use case.

We believe that the LC-CMS use case helps demonstrate that it is possible to continually run models at low resolution at global scale in a repeated manner and at low costs, making cost-effective continuous monitoring a reality. Additionally, it helped steer future efforts on other use-cases in the right direction.

For the Map Making use case, we focused on water features. The objective was to automate the detection of map-ready LC features and ingest them into the map. These features can then be used in TomTom's core map for land cover display and navigation purposes.

As part of the Map Making use case, we conducted two parallel work streams, resulting in two types of demonstrations.

First, we applied the learnings from the LC-CMS use case and fine-tuned our techniques to predict water bodies from sensor data. This involved band selection, model selection, gap filling for narrow water bodies and change detection at scale. The experiments were carried out in Myanmar to fine tune the pipeline for water prediction. For demonstration purposes, we chose six additional country sized AOIs as mentioned in table 2.4. The results were predicted for 2019 at 10m resolution. For change detection we used the Iraq & Syria region where changes were detected for drying lakes from 2019 to 2022 using the CD pipeline, drilled down from 120m to 10m. Roughly 85% of predictions at 10m were reused from 2019, with just an additional cost of processing lower 120m resolution data for huge region. This results in a huge input data cost and operational cost saving. The generated water predictions for all the AOI regions mentioned in table 2.4 are shared in the Map Making viewer application.

Second, we established a map making process where these detected water bodies are ingested into the map. We vectorized, smoothed and post-processed the water bodies generated by our Water prediction pipeline using the FME workbenches defined for Map Making. We then ingested the whole of the Chile region into TomTom maps using the TomTom ingestion process explained in D5.4 Map Making use case. The results of all the AOIs except South Korea have been shared as GeoPackage files which contain the vectorized, smoothed and map-ready vector features. Due to security concerns results for South Korea cannot be shared.

The Map Making viewer also shows a part of Chile which shows on the TomTom map before and after the water features generated from the Map Making pipeline were ingested into the TomTom maps. Map Making viewer is available to the public, therefore whole region is not shown.

D5.8 Validation report discusses the realized benefits of Map Making and LC-CMS.

The research and results of the Built-up area use-case within the GEM project have been concisely presented. The work in the use-case involved integrating existing (machine learning, EO data and processing) knowledge with the GEM platform (eo-learn and eo-grow) to establish a process for identifying new urban areas on a large scale. A 120 m resolution global cloudless mosaic Analysis Ready Data cube was created to support cost-effective monitoring. The process for detecting large built-up areas was developed, allowing for the identification of individual buildings by increasing the resolution progressively. Ground truth data from TomTom and expertise from TomTom and TUM were essential.

The Built-up use-case was the first in the series of GEM use-cases, and is considered foundational within the GEM project, as the scalable and cost-efficient approaches developed and used within were later applied also in other use-cases (e.g., LC-CMS, Map Making and Conflict Pre-Warning use-cases), and also in continuous monitoring.

Service for running large-scale, cost-effective, and Continuous Monitoring has also been presented in this deliverable. The Continuous Monitoring service aimed to explore the utilization of the GEM framework for ongoing monitoring, with a specific focus on water monitoring in the Sahel region. This involved establishing a historical baseline and continuously assessing NDWI anomalies using the Sentinel-2 observations. While the achieved results in NDWI anomalies were not exceptional, we consider the primary objective to be met. One notable aspect of the results is the flexibility of the approach facilitated by eo-grow, allowing for easy adjustments to pipelines or configuration files to incorporate different models and monitor different things. Furthermore, the service's adaptability extends to the use of various datasets, where modifying the download pipeline configuration could facilitate straightforward transitions to other sources like Sentinel-1 or Harmonized LandSat Sentinel for an extended historical perspective.

Finally, the approaches, algorithms, and tools, developed to support Built-up and other use-cases are already proving highly valuable within Sinergise. The eo-grow pipelines we have developed for the Built-up use-case are now a used daily in both our research and production work, further showing that the technology readiness level of the GEM framework is very high.

Despite somewhat modest results of the Built-up use-case, they nevertheless show that low-resolution scan already provides value, and are driving the cost-efficient drill-down approach. Using this "continuous scan mode" at low resolution, one can switch to "spot mode", employing the higher resolution data. Finally, the single ease-to-use gateway for accessing all kinds of data (from open data like Sentinels and LandSat to very-high-resolution data from Maxar, Airbus and Planet) with Sentinel Hub unified API, brings significant advantage to the GEM framework.

The Crop Identification use case demonstrator was only developed to an early stage as we could not improve crop identification accuracy with weather data. The temperature downscaling demonstrator was developed to product maturity in form of live hyper-resolution (10 m) city temperature maps on the meteoblue website. Computing climate variables on grids simplifies the generation of training data sets, as does the computation

of gridded climate indices and other derived variables. The measurements data interface allows meteoblue to manage measurement data rather than just ingesting them from 3rd parties and to internally streamline operations. This opens new possibilities such as quality assurance as a service.

A number of demonstrators are available to try out working with climate data. Depending on the maturity of the underpinning technology, they are available at different technology readiness levels (TRLs). The mature demonstrators can be used for actual work.

Regarding the Conflict Pre-Warning use case, as discussed through the deliverable, the datasets, processing framework and services developed through the GEM project are of added-value for the understanding Climate-Security issues. In particular, the cost optimized continuous monitoring has the potential to enable anticipating changes that may lead into risks for local populations. Beyond that, ad-hoc processing services also have the potential to improve existing procedures, especially in terms of data integration and fusion, such as providing situational awareness information or producing detailed analyses. These aspects have been demonstrated through the CPW use case presented in this deliverable. The CPW use case represents the first ambitious step into the setup of products and services supporting further understanding of Climate-Security issues at SatCen.