Palacký University Olomouc, Faculty of Science,

Department of Geoinformatics

Paris Lodron University Salzburg, Faculty of Natural Sciences,

Department of Geoinformatics

VISUALIZATION OF MULTI-TEMPORAL INFORMATION FROM SEMANTIC EARTH OBSERVATION DATA CUBE

Diploma thesis

Author

Rabin OJHA

Supervisor (Palacký University Olomouc)

RNDr. Jan BRUS, Ph.D.

Co-supervisor (Paris Lodron University Salzburg)

Assoc. Prof. Dr. Dirk TIEDE

Erasmus Mundus Joint Master Degree Programme
Copernicus Master in Digital Earth
Specialization Track Geovisualization & Geocommunication
Olomouc, Czech Republic, 2023

With the support of the Erasmus+ Programme of the European Union

ANNOTATION

The production and availability of multi temporal earth observation data have increased significantly mainly due to the data distribution policies and development in the technology. New ways of handling big earth observation data through data cubes and semantic data cubes have come to light. Visualization can play a key role to get information and eventually knowledge from the crude data. This thesis aims to develop a workflow for pre processing the information obtained from semantic earth observation data cube and visualizing that information leveraging the potential of different web technologies. The interactive components of the developed web application will facilitate users to manipulate the contents and explore the dynamics of the information presented. The workflow developed as part of this thesis can be used to visualize any other similar kind of multi temporal information.

KEYWORDS

spatio-temporal, remote sensing, multi-dimensional data, geovisualization, web mapping, interactivity

Number of pages 46

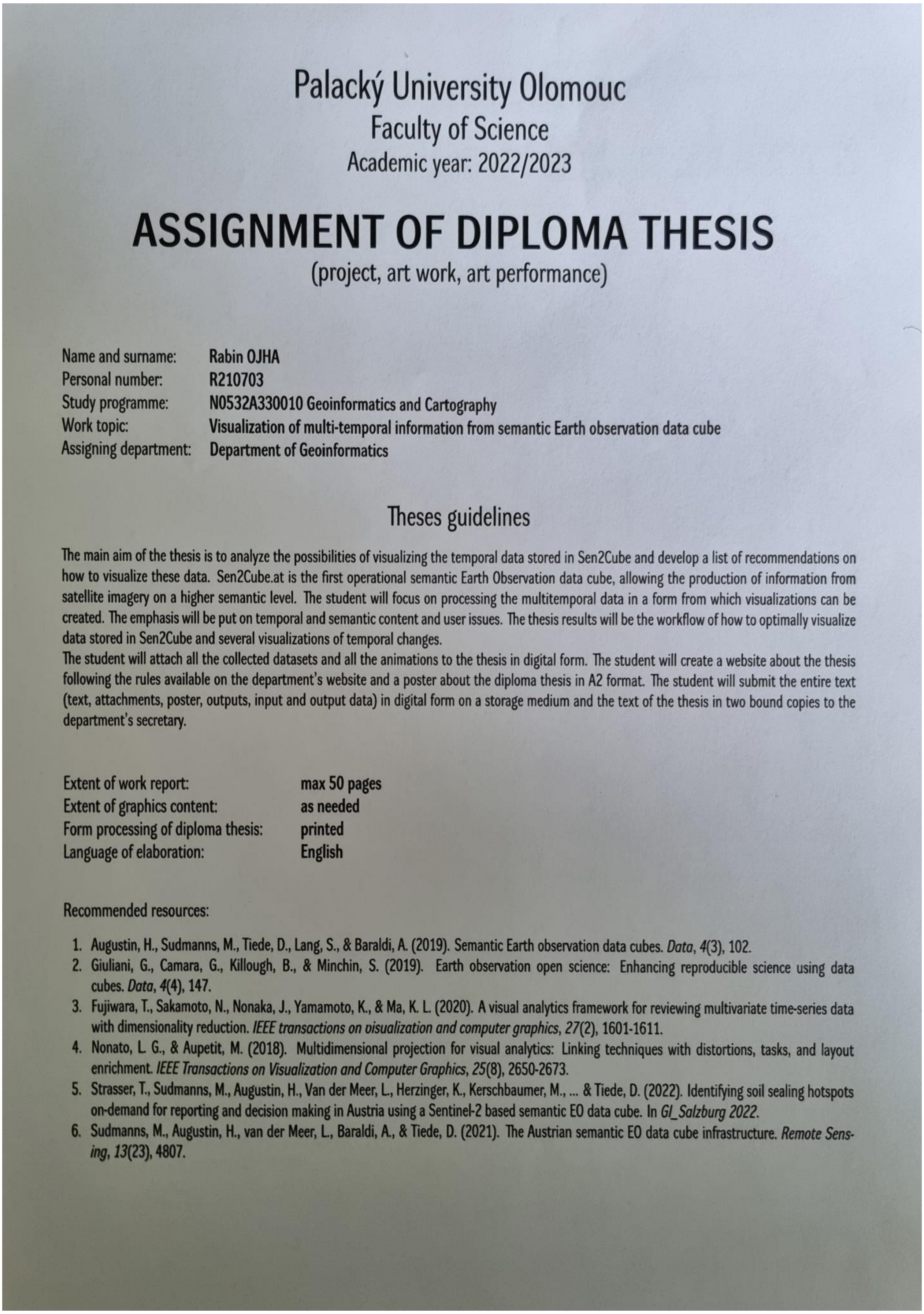
Number of appendixes 3

This thesis has been composed by *Rabin Ojha* for the Erasmus Mundus Joint Master's Degree Program in Copernicus Master in Digital Earth for the academic years 2021/2022 and 2022/2023 at the Department of Geoinformatics, Faculty of Natural Sciences, Paris Lodron University Salzburg, and Department of Geoinformatics, Faculty of Science, Palacký University Olomouc.

Hereby, I declare that this piece of work is entirely my own, the references cited have been acknowledged and the thesis has not been previously submitted to the fulfilment of the higher degree.

24.05.2023 Olomouc, Czech Republic Rabin Ojha

I am deeply grateful to my supervisors, Jan Brus (UPOL) and Dirk Tiede (PLUS), for their invaluable guidance right from the idea formulation process to the completion of this work. I would also like to extend my thanks to Martin Sudmanns and Hannah Augustin for their regular guidance. Special mention to my CDE colleagues, lecturers, and participants of the survey for user evaluation of my application. Last but not least; I would like to express special gratitude to my family and friends for always having my back.



Palacký University Olomouc

Faculty of Science

Academic year: 2022/2023

ASSIGNMENT OF DIPLOMA THESIS

(project, art work, art performance)

Name and surname:	**Rabin OJHA**
Personal number:	**R210703**
Study programme:	**N0532A330010 Geoinformatics and Cartography**
Work topic:	**Visualization of multi-temporal information from semantic Earth observation data cube**
Assigning department:	**Department of Geoinformatics**

Theses guidelines

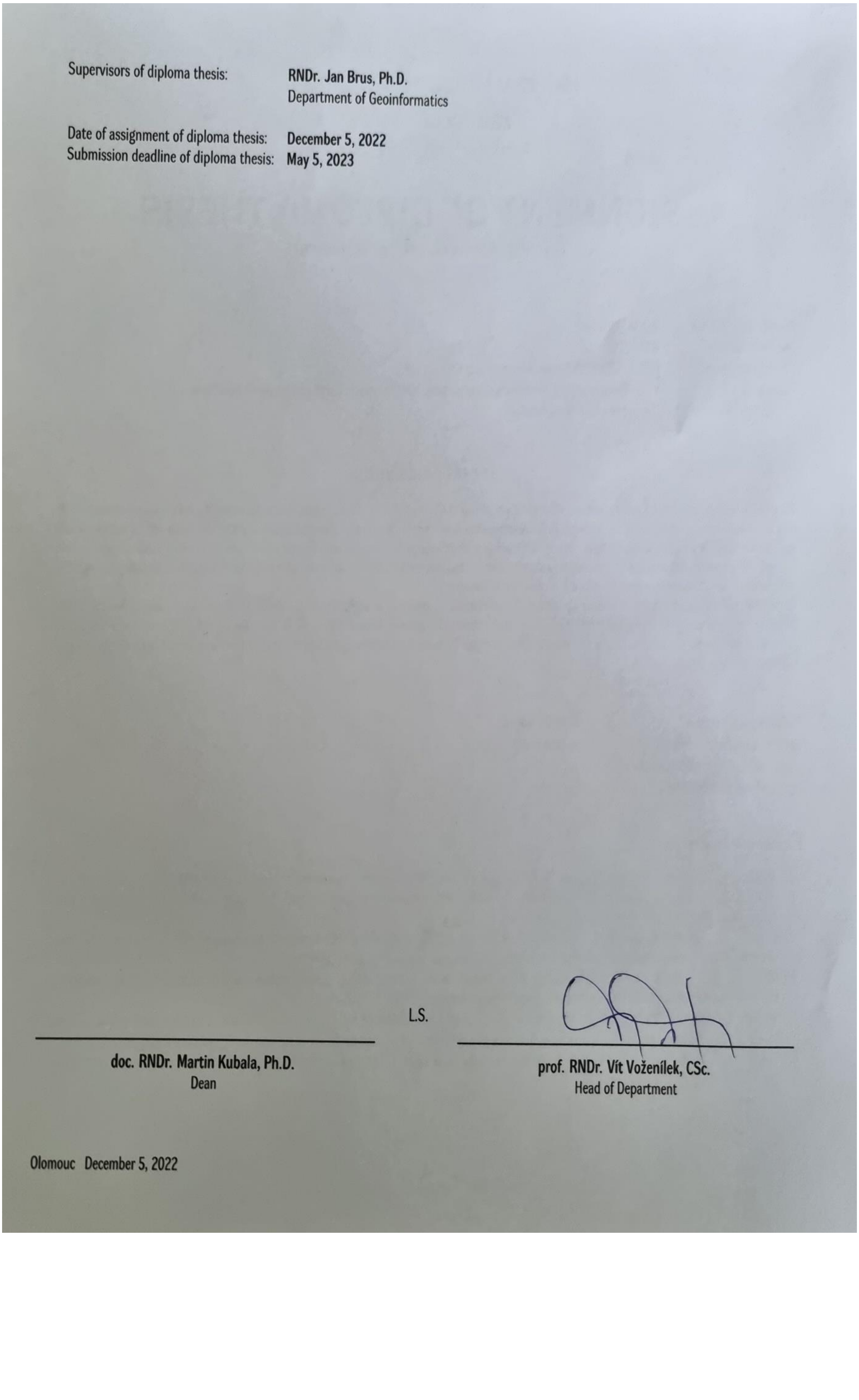
The main aim of the thesis is to analyze the possibilities of visualizing the temporal data stored in Sen2Cube and develop a list of recommendations on how to visualize these data. Sen2Cube.at is the first operational semantic Earth Observation data cube, allowing the production of information from satellite imagery on a higher semantic level. The student will focus on processing the multitemporal data in a form from which visualizations can be created. The emphasis will be put on temporal and semantic content and user issues. The thesis results will be the workflow of how to optimally visualize data stored in Sen2Cube and several visualizations of temporal changes.

The student will attach all the collected datasets and all the animations to the thesis in digital form. The student will create a website about the thesis following the rules available on the department's website and a poster about the diploma thesis in A2 format. The student will submit the entire text (text, attachments, poster, outputs, input and output data) in digital form on a storage medium and the text of the thesis in two bound copies to the department's secretary.

Extent of work report:	**max 50 pages**
Extent of graphics content:	**as needed**
Form processing of diploma thesis:	**printed**
Language of elaboration:	**English**

Recommended resources:

1. Augustin, H., Sudmanns, M., Tiede, D., Lang, S., & Baraldi, A. (2019). Semantic Earth observation data cubes. *Data*, *4*(3), 102.
2. Giuliani, G., Camara, G., Killough, B., & Minchin, S. (2019). Earth observation open science: Enhancing reproducible science using data cubes. *Data*, *4*(4), 147.
3. Fujiwara, T., Sakamoto, N., Nonaka, J., Yamamoto, K., & Ma, K. L. (2020). A visual analytics framework for reviewing multivariate time-series data with dimensionality reduction. *IEEE transactions on visualization and computer graphics*, *27*(2), 1601-1611.
4. Nonato, L. G., & Aupetit, M. (2018). Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics*, *25*(8), 2650-2673.
5. Strasser, T., Sudmanns, M., Augustin, H., Van der Meer, L., Herzinger, K., Kerschbaumer, M., ... & Tiede, D. (2022). Identifying soil sealing hotspots on-demand for reporting and decision making in Austria using a Sentinel-2 based semantic EO data cube. In *GI_Salzburg 2022*.
6. Sudmanns, M., Augustin, H., van der Meer, L., Baraldi, A., & Tiede, D. (2021). The Austrian semantic EO data cube infrastructure. *Remote Sensing*, *13*(23), 4807.



Supervisors of diploma thesis: RNDr. Jan Brus, Ph.D.
Department of Geoinformatics

Date of assignment of diploma thesis: December 5, 2022
Submission deadline of diploma thesis: May 5, 2023

L.S.

doc. RNDr. Martin Kubala, Ph.D.
Dean

prof. RNDr. Vít Voženílek, CSc.
Head of Department

Olomouc December 5, 2022

CONTENT

LIST OF ABBREVIATIONS

Abbreviation	**Meaning**
API	Application Program Interface
CSS	Cascading Style Sheets
DN	Digital Number
EO	Earth Observations
ESA	European Space Agency
FOSS	Free and Open Source Software
GDAL	Geospatial Data Abstraction Library
GeoTIFF	Geographic Tagged Image File Format
GIS	Geographic Information System
HTML	HyperText Markup Language
JS	JavaScript
ODC	Open Data Cube
OGC	Open Geospatial Consortium
PNG	Portable Network Graphics
SLD	Styled Layer Descriptor
URL	Uniform Resource Locator

INTRODUCTION

The production and availability of earth observation data have surged in the past two decades due to the introduction of free and open data policies in their distribution and technological advancements in terms of data acquisition, analysis and storage. We have come a long way since the launch of Landsat-1 in 1972. Humongous amount of (multi-temporal) EO data has been produced so far. The traditional approaches of handling EO data is being replaced by the use of 'data cubes', a multi-dimensional array based concept that provides analysis ready data to the users. The pixels in itself do not have any meanings on their own. Semantic EO data cubes go one-step ahead of the data cube by associating some meanings from the real world to each observation in the data.

Human beings are more adaptable to recognize visual patterns. A good visualization can convey complex information to users in a simpler way. It helps users to understand the depth and nuances of the data. Maps are the most common form of representing spatial information. Web technologies like HTML, CSS and JavaScript are often combined with other technologies to create visualization on the web. The trend of visualizing geospatial information on the web in the form of maps and/or other similar visualization is growing for some time now. In computer or other digital devices, RGB colour space is the most commonly used colour model. Different colours can be created by mixing three primary colours Red, Green and Blue. When rasters containing information in multiple bands are represented in softwares like QGIS and ArcGIS Pro, three primary colours will represent information in three bands and mixed colours will represent information common to two or more bands. With multi-temporal EO data, usually there is information contained in more than three bands. Representing the complete temporal information in this manner is a challenge. In regard to this problem, the thesis tries to explore the possibilities of combining both desktop and web based technologies to visualize the multi-temporal information obtained from semantic EO data cube. But, the multi-temporal information from any other source can be visualized in a similar manner.

1 OBJECTIVES

The main aim of the thesis is to analyze the possibilities of visualizing the temporal data stored in Sen2Cube and develop a list of recommendations on how to visualize these data. Sen2Cube.at is the first operational semantic Earth Observation data cube, allowing the production of information from satellite imagery on a higher semantic level. The multi-temporal information produced from the semantic data cube will pre processed in a form from which visualizations can be created. The results of the thesis will be the workflow on optimally visualizing the temporal information and web based interactive visualizations.

In the first part, the author will investigate the relevant technologies for processing and visualization of the information. That will include both web and desktop-based technologies. The workflow will then be designed separately for pre processing, storage and visualization methods and later integrated. The entire workflow will entail automation (with codes) as much as possible to ensure the reproducibility of the workflow for any similar kind of data or geographical space. The workflow will then be implemented in two use cases for two different study areas. The technical implementation can be summarized in three parts below:

- Information pre-processing
- Hosting the pre-processed information
- Development of the application

Lastly, user evaluation will be performed to test the usability of the application. Overall, the thesis will present a reproducible approach to visualizing multi temporal information leveraging the potential of free and/or open-source technologies.

2 METHODOLOGY

The chapter describes the overview of the study area, data, technology stacks, methods and processes involved in achieving the objectives of the research. A detailed description of the data, methods and procedures will be described in the chapters to come.

2.1 Study areas

The study areas for the thesis are the state of Salzburg in Austria, also known as Land Salzburg and the Syria-Turkey border region in northwest Syria.

Land Salzburg or the state of Salzburg is one of the nine states of Austria. It is located in the western part of the country with some part bordering to the Bavaria region of Germany. Salzburg is famous for its Alps and other natural landscapes. Data from Global Forest Watch shows that Salzburg had 70.9 kilo hectare in 2010. The base layer from OpenStreetMap (OSM) in Figure 1 also shows how rich Salzburg is in terms of vegetation. The area is of Salzburg is approximately 7160.5 square kilometres.

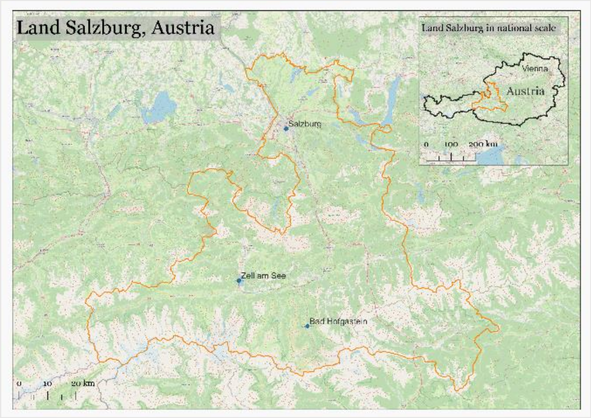


Fig 1: Study area 1

The second study area is located in the north-western region of Syria bordering the southern part of Turkey. The area of this study region is ~780 square kilometres. Euphrates, the longest river in Syria and in the western Asia runs through this area. The water level is decreasing in many parts of the posing threats to people depending on it. The northwest Syria is also prone to disasters like floods and earthquakes.

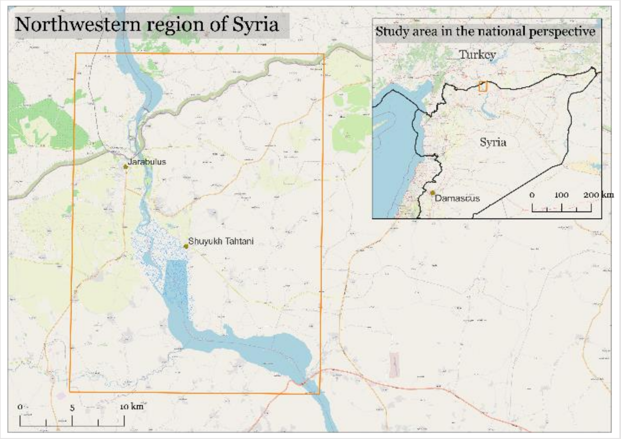

Northwestern region of Syria
Study area in the national perspective
Turkey
Syria
Damascus
0 100 200 km
Jarabulus
Shuyukh Tahtani
0 5 10 km

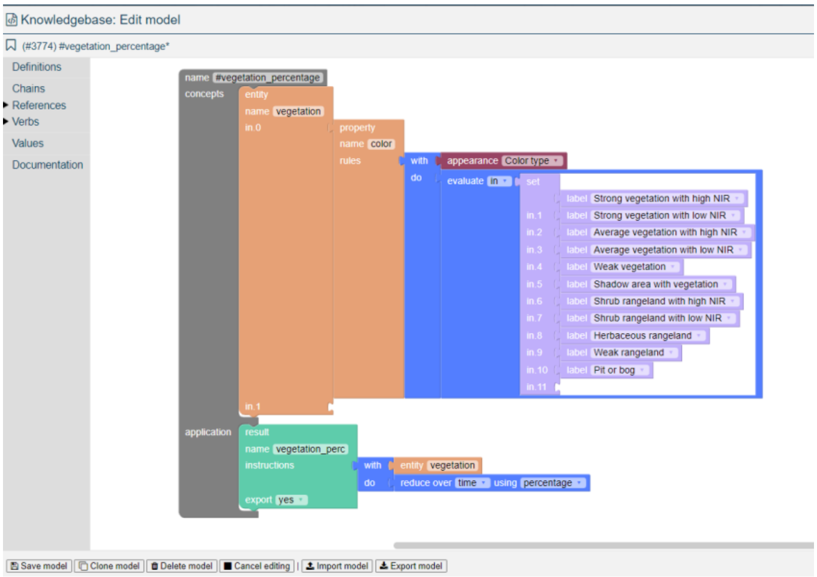
Fig 2: Study area 2

2.2 Methods

The state of the art of visualizing multi temporal data was researched. As the concept of semantic earth observation data cube and thus information obtained from it is comparatively new, the approach for visualizing information obtained from it was conceptualized. Two use cases, one related to vegetation and another related to water were defined and implemented using the latest front end technologies. Both qualitative and quantitative data based on the developed web application were collected for the user evaluation part.

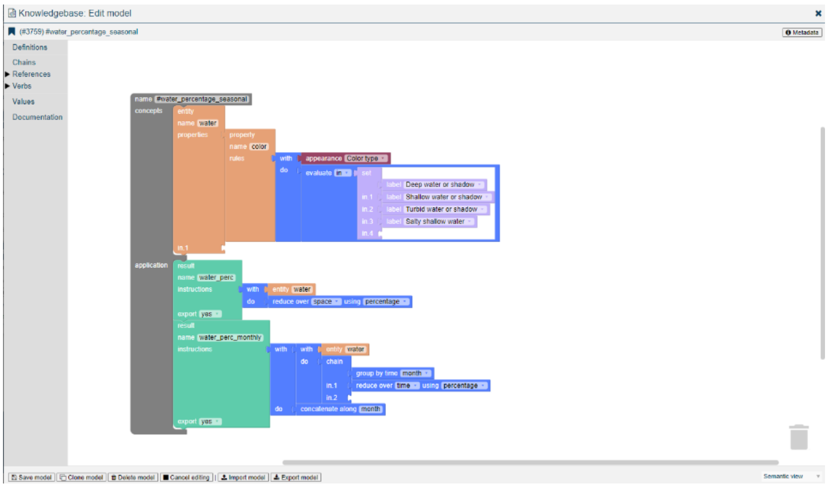
2.3 Data

Sen2Cube.at has all the Sentinel 2 scenes in its repository since the launch of Sentinel-2A in 2015 (Semantic Earth Observation Data Cube Analysis, 2023). The following two models were configured in the web interface of Sen2Cube.at to obtain information from the data in the semantic data cube.



```
name #vegetation_percentage
concepts
    entity
    name vegetation
    in.0
        property
        name color
        rules
            with  appearance Color type
            do    evaluate in
                      set
                          label Strong vegetation with high NIR
                      in.1  label Strong vegetation with low NIR
                      in.2  label Average vegetation with high NIR
                      in.3  label Average vegetation with low NIR
                      in.4  label Weak vegetation
                      in.5  label Shadow area with vegetation
                      in.6  label Shrub rangeland with high NIR
                      in.7  label Shrub rangeland with low NIR
                      in.8  label Herbaceous rangeland
                      in.9  label Weak rangeland
                      in.10 label Pit or bog
                      in.11
    in.1
application
    result
    name vegetation_perc
    instructions
        with  entity vegetation
        do    reduce over time using percentage
    export yes
```

Fig 3: Model for Salzburg region (Source: Dirk Tiede)



```
name #water_percentage_seasonal
concepts
    entity
    name water
    properties
        property
        name color
        rules
            with  appearance Color type
            do    evaluate in
                      set
                          label Deep water or shadow
                      in.1  label Shallow water or shadow
                      in.2  label Turbid water or shadow
                      in.3  label Salty shallow water
                      in.4
    in.1
application
    result
    name water_perc
    instructions
        with  entity water
        do    reduce over space using percentage
    export yes
    result
    name water_perc_monthly
    instructions
        with  with  entity water
              do    chain
                          group by time month
                    in.1  reduce over time using percentage
                    in.2
        do    concatenate along month
    export yes
```

Fig 4: Model for north-western Syria (Source: Dirk Tiede)

Approximately 2178 Sentinel 2 scenes for the Salzburg region and 140 different scenes for the north-western Syria region were used. The percentage of vegetation observed per pixel per year from 2016 to 2021 for the Salzburg region was inferred from the first model. The monthly percentage of water occurrence per pixel in the north-western Syria for the year 2022 was obtained from the second model. The information generated from both models is rasters in GeoTIFF format. In total, there are 18 rasters, 6 (yearly) for the Salzburg region and 12 (monthly) for the Syria. These rasters are going to be used for further processing, analysis and visualization in the steps ahead.

The administrative boundary for Land Salzburg was downloaded from OpenStreetMap, where the data is freely available under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF).

2.4 Software

QGIS (version: 3.28.0 Firenze), which is the most popular Free and Open Source Software (FOSS) for GIS, was used. PyQGIS (version: 3.9.5), Python Application Program Interface (API) for QGIS was used for automating the pre-processing workflows with Python scripts rather than using already established standard operations in QGIS.

Geoserver is another popular FOSS server written in Java that allows sharing of geospatial data using Open Geospatial Consortium (OGC) standards. Geoserver (version: 2.22.0) was used for hosting the raster data in the remote server at this URL https://infoviz.sen2cube.at/geoserver/web/ . A Python package, geoserver-rest (version: 1.5.1) was used to manipulate the workspaces, upload the rasters and their corresponding style files to the server. Jupyter Notebook, an open source web application, was used to document and execute the Python scripts.

For building the web application, client side languages like HyperText Markup Language 5 (HTML5), Cascading Style Sheets (CSS) and JavaScript (JS) were used. Bootstrap (version: 5.3.0), a free and open source CSS framework was used to leverage the reuse of CSS components and create responsive designs. Leaflet is one of the most popular and FOSS JavaScript mapping libraries. Leaflet (version: 1.9.3) was used for the mapping content in the web application. Jquery (version: 3.6.4), which is another FOSS JavaScript library, was used for event handling and manipulating HTML contents.

The web application is hosted on a GitHub page. Finally, the user evaluation of the web application was designed using Google Forms and disseminated to the participants.

2.5 Processing procedures

The processing procedures can primarily be divided into five steps: Pre-processing, data hosting, visualization, web application hosting and user evaluation of the web application. The waterfall model of software development was adopted where each step needed to be completed for the next step to begin. The figure below summarizes the steps involved and the technology stacks used for each step.

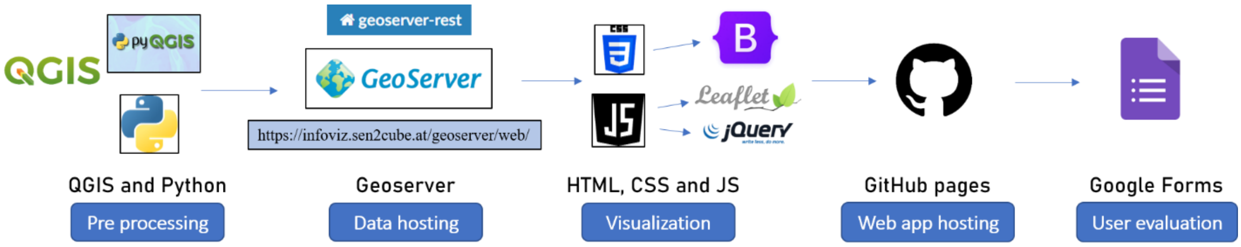


Fig 5: Workflow from data pre-processing to user evaluation of the web application

3 STATE OF THE ART

(Semantic) Earth observation data cube

Currently, the earth is continuously being observed at global, regional, national and subnational levels like never before. At the time of writing, 1662 spacecraft were launched to space as part of 'Earth Science' applications (NASA - NSSDCA - Master Catalog - Spacecraft Query, n.d.). With a large number of Earth Observation (EO) satellites in space, there is a huge production of EO data. These data have been acquired by hyperspectral, multispectral, Synthetic Aperture Radar (SAR), passive microwave, and LiDAR sensors (Ban, 2016). The introduction of free and open data policy for Landsat in 2008 (Zhu et al., 2019), for GMES (later known as Copernicus) in 2010 (Union, 2010) and other policies from several organizations have increased the accessibility of EO data and its use in scientific and societal benefits (Harris & Baumann, 2015). The enormous growth of earth observation data is also the product of rapid development in computer and remote sensing technologies (Ma et al., 2015). EO data has entered into the new paradigm of 'big data' (Guo et al., 2022) with the production of humongous amounts of data, in the range of Petabytes which is expected to reach the Exabyte scale (Nativi et al., 2015). Volume, Velocity, Variety, Veracity and Value are the five characteristics that define big data (Kalbandi & Anuradha, 2015). Thus, new mechanisms to store, manage, process and analyse them have also emerged to harness their full potential (Sudmanns et al., 2020).

EO data cubes are the new approach to handle the big data and facilitates users with Analysis Ready Data (ARD) for their works (Rizvi et al., 2018). Data cubes are arrays of multiple dimensions (Rono, 2020). EO data cubes are multi-dimensional representation of EO data with at least one non- spatial axis (Strobl et al., 2017). Data are taken from multiple sources, pre-processed, optimized and then distributed in an organized manner to the end users through different platforms. EO data are arranged in such a way that they can be accessed using spatial and/or temporal coordinates instead of using the names of the files (Augustin et al., 2019).

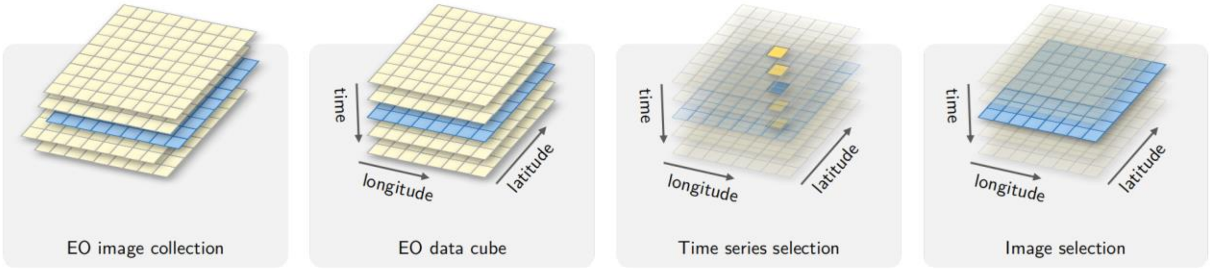


Fig 6: Concept of data cube (Source: Simoes et al., 2021)

There are several EO data cubes implementation around the world both in global as well as local level. Some of them are listed below:

- Digital Earth Australia
- Digital Earth Africa
- Austrian Data Cube
- Swiss Data Cube
- Taiwan Data Cube
- Euro Data Cube

Many of these existing data cubes and the upcoming ones are based on Open Data Cube (ODC) initiative. The ODC initiative itself stemmed out of the works of Australian Geoscience Data Cube projects. ODC aims to harness the use of satellite data for good of people, by offering free and open solution for accessing, organizing and managing large EO data (Killough, 2018).

In the normal EO data cube, pixels just have Digital Number (DN) values or pixel values that are merely a number. The pixels do not own any physical interpretation. This may impose technical constraints on some end users to utilize the EO data for their applications. Unlike the normal data cube, semantic EO data cube contains data with semantic enrichment. Semantic EO data cube refers to that data cube where each observation is mapped to at least one categorical meaning and can be queried simultaneously. For imageries, each pixel has some meaning associated with it (Augustin et al., 2019). Semantic concepts are formalized and the observations in the data cube are mapped to those semantic concepts and relationships are established. The relationship between the concepts and observations can be binary, numeric or ordinal (van der Meer et al., 2022). Users can query these formalized semantic concepts on his/her area of interest and get data in return that match those concepts.

Sen2Cube.at

Sen2Cube.at is the first implementation of semantic EO data cube infrastructure in the world and facilitates users with interactive graphical interface to perform operations on the EO data. It allows users to search and process EO data on a higher semantic level. Sen2Cube.at interface is supported by semantic EO data cube infrastructure of the same name (Sudmanns et al., 2021). Sen2Cube.at uses semantic content based image query and retrieval mechanism in the web interface to get desired information without the need of any external softwares. The user gets access to the data cube via graphical user interface which connects semantic models to information stored in the cube (Tiede et al., 2019). In addition to the graphical user interface, there is also opportunity to access Sen2Cube.at via Command Line Interface (CLI) and ArcGIS Pro Toolbox. A semantic EO data cube infrastructure intents to directly integrate computer vision at the level of EO data cube (Sudmanns et al., 2022).

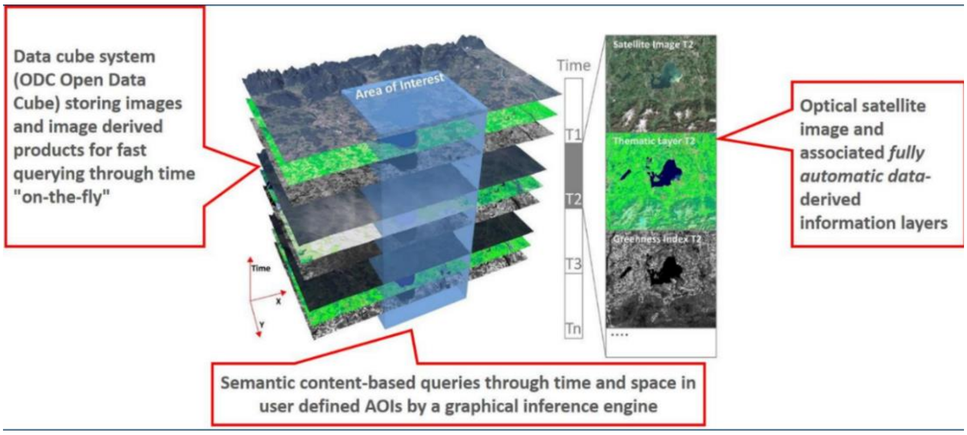
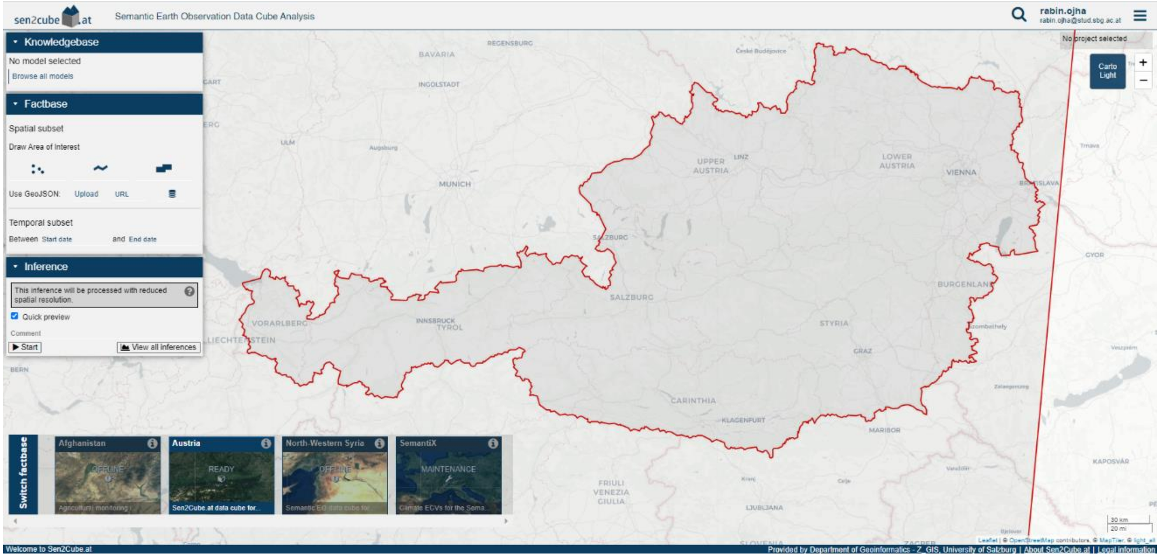

Data cube system (ODC Open Data Cube) storing images and image derived products for fast querying through time "on-the-fly"
Area of Interest
Time
T1
T2
T3
Tn
Satellite Image T2
Thematic Layer T2
Greenness Index T2
....
Time
X
Y
Optical satellite image and associated *fully automatic data-derived* information layers
Semantic content-based queries through time and space in user defined AOIs by a graphical inference engine

Fig 7: Semantic EO data cube concept in Sen2Cube.at (Source: sen2cube.at)

Figure 7 shows how the concept of semantic EO data cube is integrated in the Sen2Cube.at. Semantic queries can be made based on space and time in the area of interest defined by the users. Custom graphical semantic models can be used to create queries based on the requirement of user, without any hard code programming. The semantic models used to generate information for this thesis work can be seen in Figure 3 and Figure 4 in Chapter 2. An inference engine converts these semantic models to EO data cube friendly queries and the corresponding result is returned. The result can be downloaded in different formats optimal to be used in other GIS softwares for further works.


This inference will be processed with reduced spatial resolution.

Welcome to Sen2Cube.at

Provided by Department of Geoinformatics - Z_GIS, University of Salzburg | About Sen2Cube.at | Legal information

Fig 8: Web interface of Sen2Cube.at

At the time of this writing, the semantic EO data cube is implemented for the whole of Austria, some parts of Syria and Afghanistan and some regions of Europe. All of these factbases can be accessed from one interface as shown in Figure 8. The first three factbases are powered by Sentinel-2 sensors and contain all available imageries from 2015 to now. The last factbase, also known as SemantiX, is powered by AVHRR and Sentinel-3 A/B and contains data from 1981 to now. But the infrastructure can be implemented in any other part of the world or with any other sensors (Sudmanns et al., 2021).

Sentinel-2 mission

European Space Agency (ESA) operates the Sentinel-2 mission under their Copernicus Program. The mission consists of two satellites Sentinel-2A and Sentinel-2B, launched in 2015 and 2017 respectively. The two satellites are operated concurrently phased at 180 degrees to each other and consist of a multispectral imager with 13 spectral bands (four bands at 10 m, six bands at 20 m and three bands at 60 m spatial resolution). Sentinel-2 has three spatial resolutions of 10, 20 and 60 meters depending on the band. The temporal resolution is 10 days at the equator with a single satellite and 5 days with both. Some of the applications of products from Sentinel-2 missions are in the field of climate change, land monitoring, emergency and civil security (Sentinel-2 - Missions - Sentinel Online, n.d.).

Multi temporal data

Temporal data is data associated with time. The scale of the time could be an instant point or a longer period. Time can be expressed at different scales like days, weeks, months, years, etc. Therefore, the data can also be represented at different levels. Multi temporal data is the data of the same area collected over some time. According to (Peuquet, 1994), spatio-temporal data is a function of three components; location (where), time (when) and data itself (what). Three basic questions can be formed based on the triad framework proposed by Peuquet. Data(what) is described by when(time) and location(where). Time(when) is described by location(where) and data(what) and location(where) is described by data(what) and time(when). Time can be used to track changes in the data, finding causes of events in the past and predicting the ones in future.

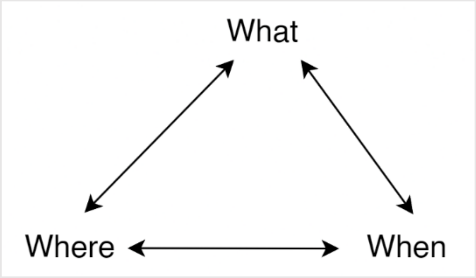


Fig 9: Triad framework of spatio-temporal data (Source: Peuquet, 1994)

The technological advancements and the open data policy from several big institutions have facilitated in the access of increasing amount of EO data. One of the active field of remote sensing for some time now is the processing of multi-temporal data (Jianya et al., 2008). Multi temporal images in EO have been used for studying different application of vegetation monitoring, trends of disasters, land surface change, etc. Remote sensing data, particularly multi temporal kind, gives us the opportunity to observe, monitor and predict the dynamics of natural and anthropogenic processes (Ban, 2016). Understanding the context in the past can help us understand the possibilities in the future. The temporal relationship in the data assists users to learn from the past, understand the current scenario and make predictions for future. Exploration and analysis of spatio-temporal data is challenging due to the need for revealing temporal dynamics in it. Visually representing this kind of data is helpful to understand it (Wu et al., 2018).

Maps and similar visualizations are the most common forms of representing spatial data. A good cartographic product or a visualization can present complex data in a simpler way to the map users. The human perceptual system is advanced and specifically suited to spot visual patterns. Visualization strives to exploit these capabilities and aids in seeing and comprehending abstract and arcane data. Unseen relationships between data and their trends can be seen through effective visualization. Interaction is considered to be an important parameter for visualizing time oriented data (Aigner et al., 2007). Static maps, dynamic maps (with certain level of dynamicity) or series of them have also been used to visualize spatio-temporal data. Interactive maps and other visual methods have also gained high popularity to visualize them (Andrienko et al., 2010).

Web mapping

Web mapping technologies have been getting popular as a medium to visualize the geospatial data. Neumann (2008) defines web mapping as the process of designing, implementing, generating and delivering maps on the World Wide Web (WWW). The three basic elements in web mapping are geodata and their visualization, software and the web (Bert et al., 2017). The advancement in technologies have drastically improved the development of web mapping technologies. Some of the key events for web mapping includes the creation of World Wide Web (WWW) in 1989, HTML in 1993, MapQuest in 1996, Geoserver in 2001, launch of Google Maps and Google Earth in 2005 and ArcGIS Online in 2012. The trend of web cartography has changed the way in which geospatial data are communicated (Cartwright, 2003).

The key distinction between designing maps for print media and for the web is that, the first one is designed for the map readers while the latter one is for map users (Muehlenhaus, 2013). Even the maps in the web can be both static and dynamic types. Both these types can be further categorized to interactive maps or view only maps. The static view only maps are the ones that are uploaded to the web after scanning the physical copies. Interactive static maps allows users to click on the map and redirect to some other information. View only dynamic maps are the simple animated maps with no possibility of interactions from users. Interactive dynamic maps have more possibility of manipulating the content in them (Kraak & Brown, 2001). They give more privilege to the users to customize the content in the maps. However, the level of interaction between human-map varies for each case. The modern cartography is characterized by interactivity and dynamicity of the elements present in it (MacEachren & Kraak, 2001). The cartographic products on the web are more interactive than static products. There are three elements of cartographic interaction; humans, map and technology. Users and map interact with each other through the medium of some technology. Both map users and cartographic products possess the capacity to induce change in one another (R. E. Roth, 2012).

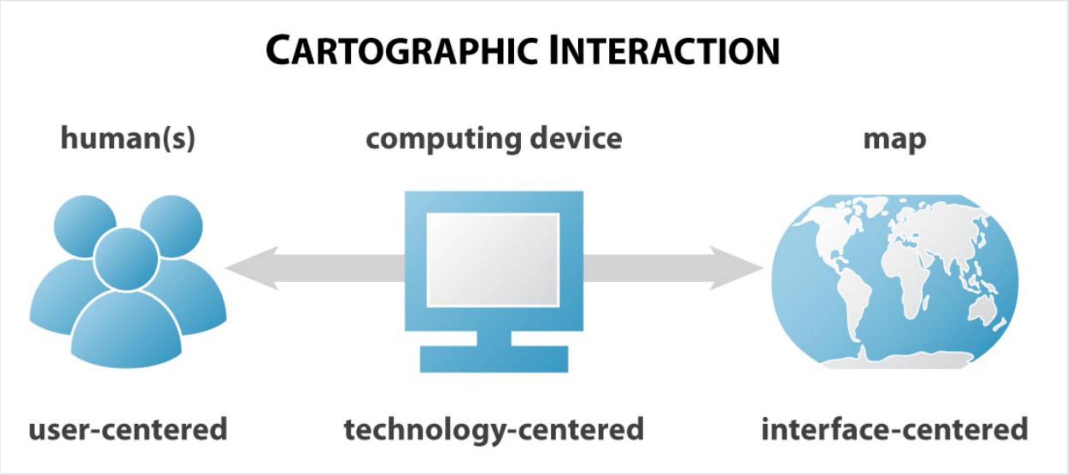


Fig 10: Elements of cartographic interaction (Source: R. Roth, 2011)

User evaluation

The use of web map applications is not always simple. Although the map is the dominant component, there are many other components involved to build a web map application. The mix of all these components might make the overall web app complex for the users to use and understand the contents in it. According to the International Standard Organization (ISO) 9241, Usability is defined as "the effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments". An evaluation mechanism is required to test the usability of the map(s) and other components of the web map application. Web map applications or any other visual analytic tools have to be evaluated by the users to test their design, interactivity and the visualization aspect (Popelka et al., 2019). Subjective user evaluation is one of the simplest ways to evaluate a product. The quality of this kind of method is highly correlated to the users participating in the evaluation. The result might not always be suitable for further interpretation due to its subjective nature (Sterba et al., 2014). Methods like surveys (online and offline) are a quick way to assess the cartographic works and are less costly. Again, it is highly dependent on the selection of the users. The semantic differential approach measures people's attitudes via semantics. Users might be asked if they like or dislike something if they found something attractive or dull or similar (Bláha, 2009). The Likert scale is a popular psychometric technique to measure the attitude of participants towards the question (Joshi et al., 2015). There are other sophisticated methods like eye tracking and application-based methods for user evaluation that provides very accurate results.

4 DEVELOPMENT PROCESS OF WEB APPLICATION

The chapter explains all the processes involved in the development of the web application starting from the pre-processing of the information obtained from Sen2Cube.at to hosting the web application on a GitHub page in detail.

4.1 Information pre processing

Six individual rasters, one for each year from 2016 to 2021 were obtained from the first model in Sen2Cube.at. Each raster contained a percentage of vegetation observed per pixel for Land Salzburg in that particular year. The vegetation distribution for 2016 looked as follows:

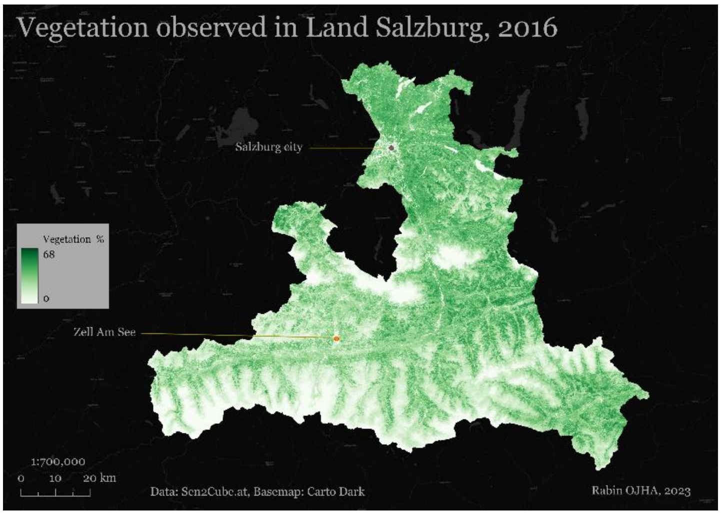

Vegetation observed in Land Salzburg, 2016
Salzburg city
Vegetation %
68
0
Zell Am See
1:700,000
0 10 20 km
Data: Sen2Cube.at, Basemap: Carto Dark
Rabin OJHA, 2023

Fig 11: Percentage of vegetation observed for Land Salzburg in 2016

As mentioned in the last chapter, each pixel in the semantic EO data cube is mapped to at least one categorical interpretation. The interpretation could be anything like water, vegetation, etc. Percentage in this case represents the occurrence of 'vegetation' in that particular pixel per 100 available imageries during the year. The range of percentage varies from year to year as shown in the figure below.

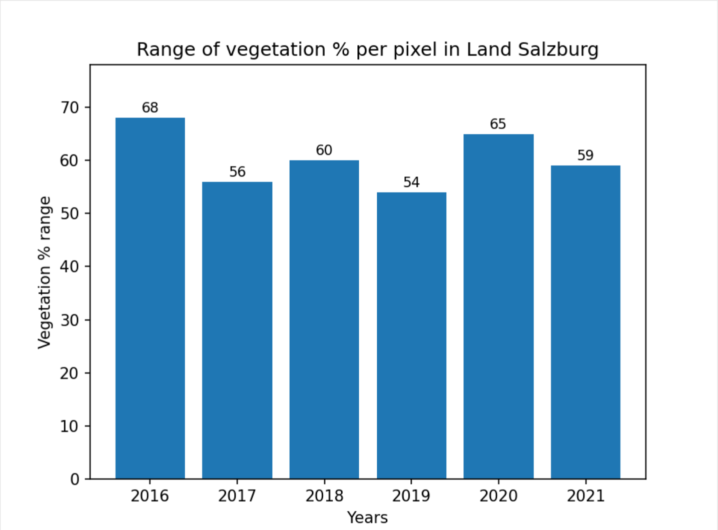


Fig 12: Variation of vegetation percentage observed in different years

From the second model, a single raster with 12 bands, each containing the monthly percentage of water occurrence per pixel for the study area in the north-western Syria region for 2022 was obtained. The single multiband raster was converted to multiple single bands rasters in Python using the *gdal_translate* program of the Geospatial Data Abstraction Library (GDAL). Subsequently, 12 rasters were obtained, one for each month of 2022. The monthly distribution can be illustrated in the figure below.

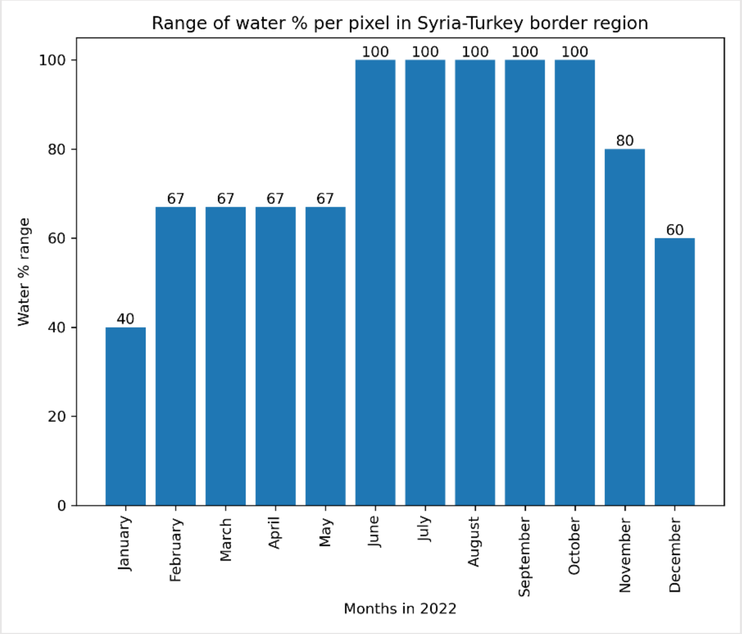


Fig 13: Variation of water percentage observed in different months of 2022

The collection of rasters for the use case in Salzburg and the Syria were loaded separately in QGIS for the next steps.

4.1.1 Raster thresholding

The first step was to create binary rasters based on certain threshold percentages. The threshold percentages for vegetation in Salzburg were set to 10, 20, 30, 40 and 50 considering the range of vegetation percentage over the studied years. For each threshold, a Python script generated six binary rasters, one for each year from 2016 to 2021. The binary rasters created for that threshold contained only those pixels with value equal to or greater than the threshold value. The total number of rasters created was 30. The binary values in resulting rasters were programmed in a geometric sequence of two as shown in the table below.

Table 1 Years and their corresponding values in resulting raster after raster thresholding

Years	**Binary values in resulting raster**
2016	0 or 64
2017	0 or 32
2018	0 or 16
2019	0 or 8
2020	0 or 4

2021	0 or 2

Similarly, the script was run for the percentage of water occurrence. The threshold percentage, in this case, was set to 10, 20, 30, 40, 50, 60 and 70, again considering the range of monthly water percentage in 2022. For the case of water in Syria, 12 binary rasters were created for each threshold percentage which in total adds to 84 rasters. The values in the binary rasters were programmed in the same way as in the vegetation case of Salzburg but with 12 orders.

Table 2 Months and their corresponding values in resulting raster after raster thresholding

Months	**Binary values in resulting raster**
January	0 or 4096
February	0 or 2048
March	0 or 1024
April	0 or 512
May	0 or 256
June	0 or 128
July	0 or 64
August	0 or 32
September	0 or 16
October	0 or 8
November	0 or 4
December	0 or 2

The reason for using different binary values for each year or month is to assist in the identification of which month(s) or year(s) the pixel values correspond to, in further steps.

4.1.2 Raster addition

All the newly created binary rasters for each threshold percentage were added using another Python script, which can be found in the attachment. It is to be noted that the addition operation was performed on a pixel level. The pixel values of corresponding pixels in different rasters were added. So, five rasters for five threshold values were obtained for the Salzburg region and seven rasters for seven threshold values were obtained for Syria.

4.1.3 Generation of unique values' rasters

Using the rasters obtained in the last step as input, new rasters were created with the help of another Python script that generates individual rasters based on the unique values present in the input.

```
threshold = 50
in_raster = f'path_to_directory/final_output50.tif'
```

```
with rasterio.open(in_raster) as src:
        raster_data = src.read(1)
        unique_values = np.unique(raster_data)

for val in unique_values:
out_raster = f'path_to_directory/raster_{int(val)}_{threshold}.tif'
out_ds = gdal.GetDriverByName('GTiff').Create(out_raster,
in_ds.RasterXSize, in_ds.RasterYSize, 1, data_type,
options=['COMPRESS=DEFLATE'])

out_ds.SetProjection(in_ds.GetProjection())
out_ds.SetGeoTransform(in_ds.GetGeoTransform())

data = in_band.ReadAsArray()
data[data != val] = 0
out_band = out_ds.GetRasterBand(1)
out_band.WriteArray(data)
```

The snippet of the script above shows how a raster is taken as input, the unique values in it are read and then new rasters are named and created with that unique value with same properties as input raster. The number of rasters created in this step were equal to the number of unique pixel values present in the input rasters. Around 300 unique rasters for vegetation and 10300 unique rasters for water occurrence were generated in total. The values in these rasters are binary. They either consist of the unique value or zero as their pixel values. These newly created rasters are going to be queried using front-end languages and visualized in the web application in the further steps.

The rasters created were named in such a way that it would be easier to identify and query them in the subsequent stages. For the vegetation case, rasters were named as *raster_{unique_values}_{threshold}* as seen in the code above. 'Unique_values' in the name corresponds to the binary values of one single year or sum of multiple years. 'Threshold' corresponds the threshold percentage used for binary encoding in 4.1.1. For example: raster_4_30 represents the pixels that were vegetated only in 2020 but not in rest of the years and with 30% threshold. raster_38_10 represents pixels that were vegetated only in 2017,2020 and 2021 but not in the remaining years and with 10% threshold. For the water occurrence, rasters were named as *water_{unique_values}_{threshold}* . 'Threshold' corresponds again to the threshold percentage used for binary encoding in 4.1.1 and 'unique_values' corresponds to the binary values of one single month or sum of multiple months as shown in Table 2. These unique values rasters represent only exclusive pixels for the selected years in Salzburg or months in Syria.

4.1.4 Generation of style files

Styled Layer Descriptor (SLD) is an OGC standard that controls the appearance of geospatial data ("Styled Layer Descriptor," n.d.). It is the primary styling language used by Geoserver. SLD style files were generated with Python scripting inside QGIS. Red (*HEX code: #ff0000*) and blue (*HEX code: #0000ff*) colours were used for styling the pixels for both vegetation and water occurrence based on their high visual impact or strong visual presence nature. Two set of SLD files for the all rasters were generated for each study area based on the requirement in next steps. Black (*HEX code: #000000*) and white (*HEX code: #ffffff*) colours were used for styling the boundaries in both cases. The version of SLDs exported by QGIS were 1.0 and 1.1 for raster and vector respectively.

The individual style file would be associated with the corresponding raster in Geoserver in the next steps to come. The schematic below summarizes the overall works involved in the pre processing of information obtained from Sen2Cube.at. The completion of these processes makes our raster ready to be hosted and visualized in the next steps.

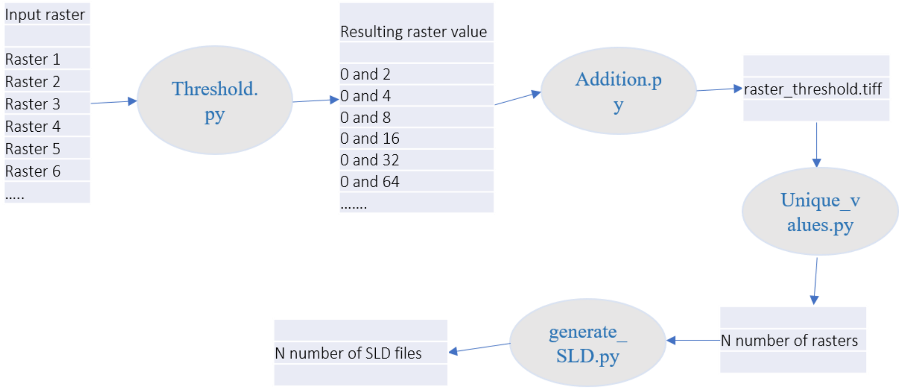


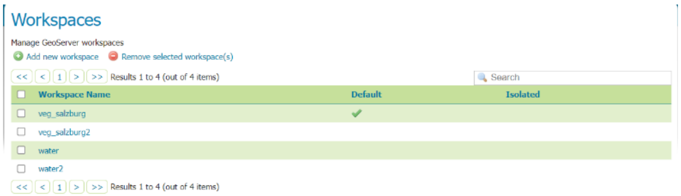
Fig 14: Illustration of overall raster pre-processing workflow

4.2 Hosting data in Geoserver

The raster and vector data were hosted in Geoserver. It is an open source server to publish data to serve in web applications (Iacovella, 2014). Workspaces, Stores, and Layers, which are three of the fundamental data types in Geoserver, were used for hosting the data. A layer is a vector or raster data that is stored in Geoserver. A store is a medium to connect data to its source. Finally, workspaces are containers to group items together. For example: Layers and style files can together be in a workspace.

4.2.1 Creation of workspace

Four workspaces were set up in Geoserver, two for vegetation in Salzburg and another two for water in the Syria region. The workspaces were named as veg_salzburg, veg_salzburg2, water and water2 as shown below. The first two were set up for the vegetation in Salzburg and the other two were set up for water occurrence in north-western Syria.



Workspaces

Manage GeoServer workspaces

Results 1 to 4 (out of 4 items)

□	Workspace Name	Default	Isolated
□	veg_salzburg	✓	
□	veg_salzburg2		
□	water		
□	water2		

Results 1 to 4 (out of 4 items)

Fig 15: Screenshot of the Geoserver displaying workspaces

The reason for creating two workspaces for the same region was to be able to style the same rasters with two different colours (red and blue) in the next steps and be able to use it for comparison map in the web application developed in the later stage.

4.2.2 Upload of data and styles

The rasters pre processed for vegetation in Salzburg were uploaded to two workspaces veg_salzburg and veg_salzburg2. Two different sets of the same rasters organized in two different folders in the local directory were uploaded to these two workspaces. SLD files containing red colour were uploaded to the first workspace and the ones with blue colour were uploaded to the second workspace. Boundary files as well as the SLD files to be used with them were also uploaded to the workspace. Considering the large volume of both rasters and style files, a Python package called "geoserver-rest" was used to automate the entire process from the creation of the workspace to the upload of data and SLD files to Geoserver. The geoserver-rest is an open source Python package to manage data in Geoserver (Kshetri et al., 2021). A similar process was repeated for another two workspaces 'water' and 'water2' for the case of water occurrence in north-western Syria.

4.2.3 Linking of data and styles

Finally, the SLD files in the workspace were associated with the corresponding files (both raster and vector) in the same workspace. The snippet of the Python script below shows the uploading of style files to one workspace and associating them with the rasters already in that same workspace with the help of 'geoserver-rest'.

```
import glob
import os
from geo.Geoserver import Geoserver
geo = Geoserver('https://infoviz.sen2cube.at/geoserver',
username='rabin-admin', password='*************')
sld_directory= "path_to_directory/sld_folder"

# Uploading SLD files
for filename in os.listdir(sld_directory):
    if filename.endswith('.sld'):
        filepath = os.path.join(sld_directory, filename)
        geo.upload_style(path=filepath, workspace='veg_salzburg')

# Associating the uploaded SLD files with rasters in the workspace
```

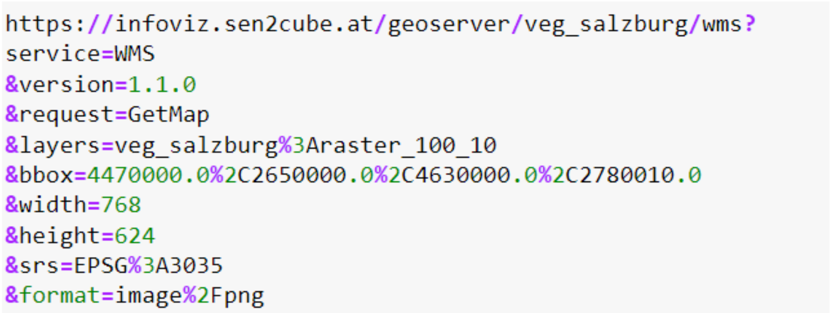
```
for filename in os.listdir(sld_directory):
    if filename.endswith('.sld'):
        filepath = os.path.join(sld_directory, filename)
        layer_name = os.path.splitext(filename)[0]
        style_name = os.path.splitext(filename)[0]
        geo.publish_style(layer_name=layer_name,  style_name=style_name,
workspace='veg_salzburg')
```

The code imports necessary packages including geoserver-rest, log into Geoserver with credentials and sets the directory for style files in the first block. The second block uploads the style files with .SLD extension from the local directory to the 'veg_salzburg' workspace. Finally, the uploaded SLD files are matched with the rasters of the same name in the workspace. Finally, individual SLD files are associated with individual raster layers, which already exist in the same workspace, in the last block of the code. It is to be noted that the interrelated raster layer and SLD file share the same name, which is what helps to match them together. The same phenomenon was repeated for three other workspaces but with a different set of SLD files.

4.2.4 Publishing of Web Map Service (WMS) images

Geoserver supports different OGC standards like Web Map Service (WMS), Web Feature Service (WFS), Web Coverage Service (WCS), etc. for serving the data. The type of data to be served is one of the factors to define the kind of standard to be used for serving. WMS produce maps of spatial data in image formats like JPEG, PNG, GIF, etc (de la Beaujardiere, 2006). WMS also returns images in other formats like PDF, KML, GeoRSS, etc. The other formats that WMS can return images are PDF, KML, etc. (WMS Output Formats — GeoServer 2.24.x User Manual, n.d.). The WMS service for all four workspaces was enabled. WMS was chosen over WCS to serve the rasters because no operations were to be needed for the rasters. All of the rasters were already pre-processed in the last step and only needed to be displayed as an image. The GetMap operation of WMS would be used in the next steps to request the Geoserver to generate the map image in Portable Network Graphics (PNG) format.

A typical WMS request for layer 'veg_salzburg:raster_100_10' in Coordinate Reference System (CRS) of EPSG:3035 in PNG format with default style looks like the following. The height, width and the bounding box of the request image are also specified.



```
https://infoviz.sen2cube.at/geoserver/veg_salzburg/wms?
service=WMS
&version=1.1.0
&request=GetMap
&layers=veg_salzburg%3Araster_100_10
&bbox=4470000.0%2C2650000.0%2C4630000.0%2C2780010.0
&width=768
&height=624
&srs=EPSG%3A3035
&format=image%2Fpng
```

Fig 16: Example for GetMap operation of WMS

In summary, rasters, vectors and style files were uploaded to Geoserver, and geospatial data and their corresponding styles were linked to each other. And they were prepared to be served with WMS as images and thus visualized in web application in successive stages.

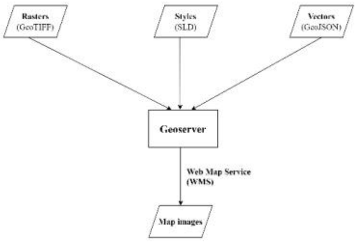


Fig 17: Schematic representation of overall workflow of preparing data for next step

4.3 Visualization

Considering interactivity as the main factor and thus providing users with more rights to the map products, the decision to create an interactive web application was made. The client-side web application to visualize the rasters created in previous steps was developed using HTML, CSS and JS. This section deals with the processes, principle concepts, libraries and other components used to develop the web application. This step was initiated after the completion of hosting data in the server.

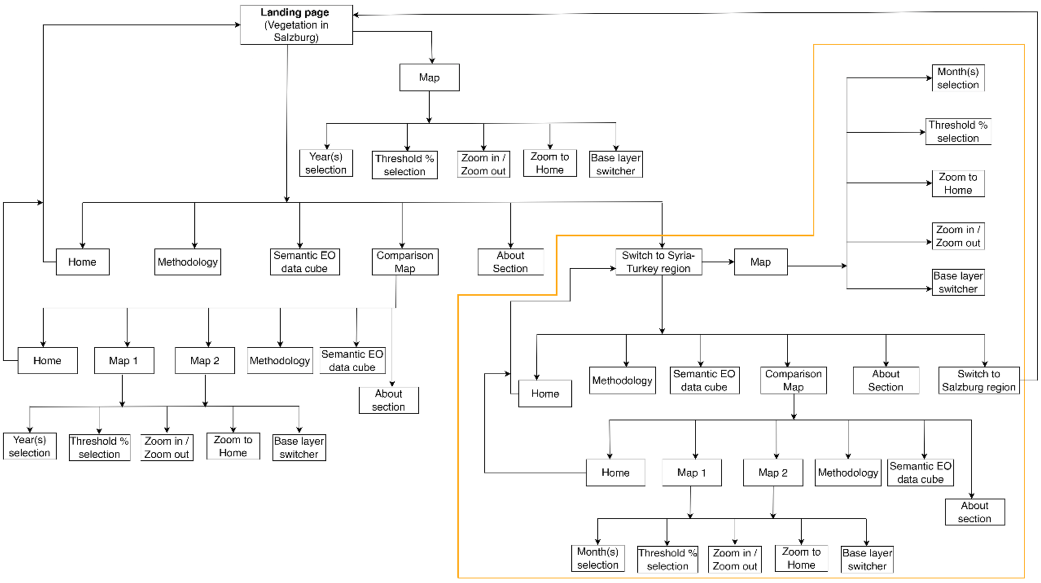
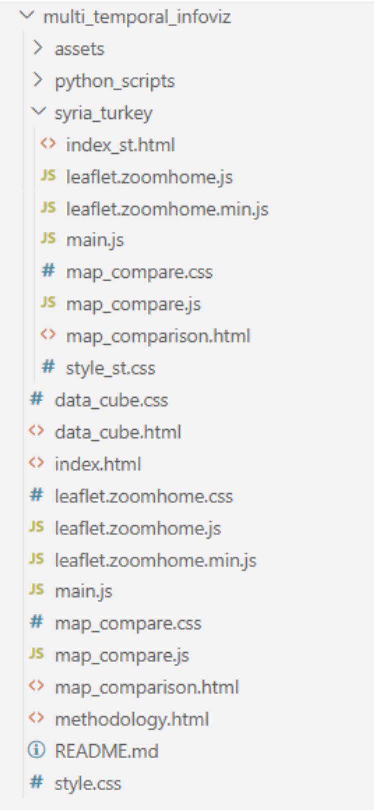


Fig 18: Architecture of the client-side visualization

The web application primarily consist of navigation bar and cartographic visualization. The visualization covers the significant portion of the layout and is the main component of the application. The navigation bar is to assist the readers with additional information and switching study areas. The 'Methodology', 'Semantic EO data cube', and 'About' section contains information about data cube, pre processing and visualization approach that complements the map users and enhance their understanding of the map/cartographic products in the application. The vegetation case in Salzburg was set as the main landing page of the web application. Additional web pages for other sections of the application were set up. This included both HTML and CSS files for each of the web pages. The assets for the web application and web pages for another study area, north-western Syria, were set up in two separate sub-directories inside the main directory for organizing the web pages systematically. The illustration above shows the detailed schematic representation of the architecture of the visualization side of the application.

The contents and their appearance were controlled by HTML and CSS. Various pre built components like containers, modals, buttons, tooltip, navigation bar, etc. of Bootstrap, a popular CSS framework was leveraged instead of manually setting up the styles. Advantages of using Bootstrap over custom styles were the consistency of contents across different browser, time efficiency, etc. The light colours were used for background, navbar and other elements of the app while the main visualization in mapping contents were coloured with sharp colours like red and blue to create visual interest and dominance in the cartographic product. 'Roboto' is a sans serif font, developed by Google for digital media specifically for Android. The font is licensed under Apache License, version 2.0 which makes it free to use. The same font, with different weights and sizes were the only ones to be used in the entire application to keep it simple.



```
v multi_temporal_infoviz
  > assets
  > python_scripts
  v syria_turkey
    <> index_st.html
    JS leaflet.zoomhome.js
    JS leaflet.zoomhome.min.js
    JS main.js
    #  map_compare.css
    JS map_compare.js
    <> map_comparison.html
    #  style_st.css
  #  data_cube.css
  <> data_cube.html
  <> index.html
  #  leaflet.zoomhome.css
  JS leaflet.zoomhome.js
  JS leaflet.zoomhome.min.js
  JS main.js
  #  map_compare.css
  JS map_compare.js
  <> map_comparison.html
  <> methodology.html
  ⓘ  README.md
  #  style.css
```

Fig 19: Code structure as seen in Visual Studio Code

JavaScript was used to add interactivity to the web pages. Leaflet, which is a popular open source mapping library based in JS was used for the main visualization. Leaflet has limited features by default to maintain its lightweight nature. However, there are a lot of plugins that can be added to Leaflet to extend its capabilities. Two plugins in particular, Leaflet.zoomhome and Leaflet.Sync, were used in our application to create custom zoom home button and synchronize two maps during comparison respectively. The native zoom control was disabled and the plugin was used because of its better aesthetics. In addition to Leaflet, jQuery, another JS library, was used to handle events and manipulate HTML content.

```
var slider = document.getElementById("threshold_range");
var valuesDiv = document.getElementById("slider-values");
var threshold_range = $("#threshold_range").val();
$("#threshold_range").on("change", function() {
  threshold_range = $(this).val();
});
for (var i = 10; i <= 50; i += 10) {
  var span = document.createElement("span");
  span.innerText = i;
  valuesDiv.appendChild(span);
}
slider.addEventListener("input", function() {
  var value = slider.value;
  var spans = valuesDiv.getElementsByTagName("span");
  for (var i = 0; i < spans.length; i++) {
    if (spans[i].innerText == value) {
      spans[i].classList.add("active");
    } else {
      spans[i].classList.remove("active");
    }
  }
});
```

The code snippet above changes the value of threshold percentage in the backend on the change of slider in the front end by map reader and stores that value.

```
var yearSum = 0;
  years.map(function(year){
    if (year == 2021) {
      yearCode = 2
    } else if (year == 2020) {
      yearCode = 4
    } else if (year == 2019) {
      yearCode = 8
    } else if (year == 2018) {
      yearCode = 16
    } else if (year == 2017) {
      yearCode = 32
    } else if (year == 2016) {
      yearCode = 64
    }else{
      yearCode = 0
    }
    yearSum = yearSum + yearCode
  })
```

The function with year and yearCode is the core logic of our application. It shows how user queries particular years in the frontend. This code comes from Table 1 in the raster thresholding stage. A new variable is assigned based on 'yearSum' and 'threshold_range', which comes from the function described previously. The same variable name is taken by Leaflet and matched with the raster of the same name in the server. Finally, when it finds it, Leaflet adds the WMS image of the raster to the map.

```
var rasterLayerName = `raster_${yearSum}_${threshold_range}`
var wmsLayer =
L.tileLayer.wms('https://infoviz.sen2cube.at/geoserver/veg_salzburg/wms'
, {
layers: `veg_salzburg:${rasterLayerName}`,
format: 'image/png',
version: '1.1.0',
transparent: true,
zIndex: 1000
});
wmsLayer.addTo(map);
```

For the water occurrence in Syria region, the threshold values ranged from 10 to 70. The 12 months were encoded as follows in JS based on Table 2.

```
if (month == 1) {
      monthCode = 4096
    } else if (month == 2) {
      monthCode = 2048
    } else if (month == 3) {
      monthCode = 1024
    } else if (month == 4) {
      monthCode = 512
    } else if (month == 5) {
      monthCode = 256
    } else if (month == 6) {
      monthCode = 128
    } else if (month == 7) {
      monthCode = 64
    } else if (month == 8) {
      monthCode = 32
    } else if (month == 9) {
      monthCode = 16
    } else if (month == 10) {
      monthCode = 8
    } else if (month == 11) {
      monthCode = 4
    } else if (month == 12) {
      monthCode = 2
    }else{
      monthCode = 0
    }
    monthSum = monthSum + monthCode
  })
```

A variable is assigned based on 'threshold_range' and 'monthSum'. Leaflet matches the name of that variable to the raster name in the server and then displays it as a WMS image in the map in PNG format.

```
var rasterLayerName = `water_${monthSum}_${threshold_range}`
var                              wmsLayer                              =
L.tileLayer.wms('https://infoviz.sen2cube.at/geoserver/water/wms', {
      layers: `water:${rasterLayerName}`,
      format: 'image/png',
      version: '1.1.0',
      transparent: true,
      zIndex: 1000
      });
wmsLayer.addTo(map);
```

For the comparison maps in each case, the similar logic was applied but with two different maps, workspaces and layers.

4.4 Hosting the application

The entire source code and the resources used in the web application was managed using GitHub and the application was hosted in a GitHub page at this link https://rabenojha.github.io/multi_temporal_infoviz/ .

5 USER EVALUATION

The main focus of the user evaluation of the web application were the maps and the content in them. Having said that, other components of the web application were also evaluated. The objective was to assess the comprehension of map information by map users and also to collect qualitative feedbacks about the cartographic elements as well as other aspects of the web application.

A set of 25 questions, which included both subjective and objective types, were designed. Google Form was chosen to host the questionnaires because of its user-friendly interface to design and due to its popularity. The hierarchy of the questions were ordered from complex to simple. The complex to simple hierarchy allows participants to engage with more cognitive effort in the beginning and gradually phase out the cognitive load with basic questions. This helps participants from getting fatigued at the end of the survey. The questionnaires was designed in such a way that users had to navigate to almost all part of the web application. The possible answers for both subjective and objective questions were based on text and multiple choices. Some of the answers were based on five points Likert Scale.

Table 3 Survey questionnaires and their probable answers

Question number	Question	Possible answers
1,2,6	What does 'red' color in the map below represents? What information is being shown? (*Please be as specific as you can*)	*Text*
3,4	What does 'blue' color in the map below represents? What information is being shown? (*Please be as specific as you can*)	*Text*
5	What does the map in the left and right below represents? What information is being shown?	*Text*
7	'In semantic EO data cube, at least two categorical interpretations are available.' True or False?	True or False
8	What information and year is the map in Syria - Turkey region showing?	*Text*

9	Among January and February, which month has more pixels with water for 20% threshold in Syria-Turkey border region?	January, February or Not sure
10	Among February and March, which month has fewer pixels with water for 30% threshold in Syria-Turkey border region?	February, March or Not sure
11	How easy is it to navigate the overall platform?	On a scale of *1-5* 1- Easy 5- Difficult
12	Can you describe the principle used in 'About' section of the application, in very short?	*Text*
13	Were you able to understand the multi-temporal information better this way?	*Text*
14	What do you feel about the fonts used in the application?	On a scale of 1-5 1- Worst 5- Best
15	What do you feel about the colors used in the application?	On a scale of 1-5 2- Worst 5- Best
16	What do you feel about the maps and basemaps used in the application?	On a scale of *1-5* 1- Worst 5- Best
17	Do you want to add any comments on the fonts, colors, and maps used in the application?	*Text*
18	What do you like about the application?	*Text*
19	What do you dislike about the application?	*Text*

20	How familiar are you with the concept of 'Remote Sensing'?	Beginner, Intermediate, Expert, Never heard of it
21	How familiar are you with the concept of 'Earth observation data cubes and/or Semantic data cubes'?	Beginner, Intermediate, Expert, Never heard of it
22	What is your age?	15-20, 21-24, 25-30, 31-40, 41 and above
23	What is your gender?	Male, Female, Non binary, Prefer not to say, Other
24	Highest Level of Education	No formal education, High school, Bachelor's Degree, Master's Degree, PhD, Other
25	Education Background	*Text*

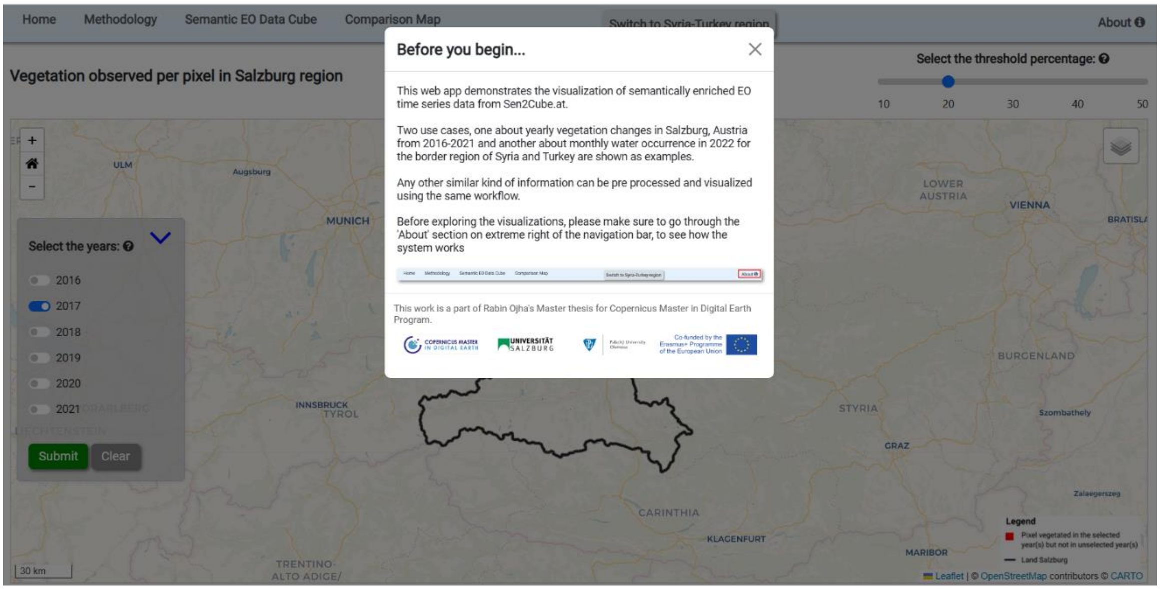
The context of the survey along with a short message, link to the web application and screenshot of the landing page of the application explaining different components was included on top of the form. The respondents were also requested to go through the application and be familiarised with it before answering the survey questions. Screenshots of the web application were attached to the question number 1, 2, 3, 4, 5 and 6. The form was disseminated to the participants through social media and responses were collected remotely. The results of the user evaluation are presented in the next chapter.

6 RESULTS

This chapter presents a comprehensive overview of the results obtained from the overall works performed in the previous two chapters. The results presented herein contribute to the overall objective of the thesis defined at the start. The chapter is divided into three sections. The first section deals with the main output, which is the web map application itself. The second section focuses on some of the insights about the data. Finally, the last section presents the findings from the user evaluation of the application.

6.1 Web application

The primary output of the workflow described in chapter three and the overall thesis is the web map application. The link for the web application is https://rabenojha.github.io/multi_temporal_infoviz/ . The web map application presents visualization of multi-temporal information processed from Sen2Cube.at with two uses focused on Salzburg, Austria and northwest Syria.



Before you begin...

This web app demonstrates the visualization of semantically enriched EO time series data from Sen2Cube.at.

Two use cases, one about yearly vegetation changes in Salzburg, Austria from 2016-2021 and another about monthly water occurrence in 2022 for the border region of Syria and Turkey are shown as examples.

Any other similar kind of information can be pre processed and visualized using the same workflow.

Before exploring the visualizations, please make sure to go through the 'About' section on extreme right of the navigation bar, to see how the system works

This work is a part of Rabin Ojha's Master thesis for Copernicus Master in Digital Earth Program.

Fig 20: Landing page of the web application

The application welcomes the user with a dialogue box giving them some context of the work as shown in the figure above. The key components of the web application are:

a. **Navigation bar**: The navigation bar is located at the top of each web page of the application and extends horizontally across the entire container. The home page consists of six navigation items listed below. The vegetation case in Salzburg is the default map in the home page but users can switch to the second case of water occurrence by clicking an item on the navigation bar.

- Home: The 'Home' item redirects user to the main web page of the respective use cases.

- Methodology: This item explains the process of visualizing multi temporal information from Sen2Cube.at briefly. This can be thought of as a gist of chapter four of this document.
- Semantic EO Data Cube: The item presents the concept of data cube and semantic data cube in very short. It was included in the navigation bar to familiarize users with the introduction of this 'new concept'.
- Comparison Map: This item redirects users to another page where users can compare two or more years in two maps side by side. More about the comparison map will be discussed in the later section of this chapter.
- Switch to Syria-Turkey region: Switching to another study area is very straightforward with a click. On clicking this item, the user will be navigated to another web page with a similar interface for water occurrence case. But if the user is already in the Syria-Turkey region, the item will change to 'Switch to Salzburg region' and clicking on it will redirect to the interface for the vegetation case.
- About: 'About' section outlines the approach used for visualization in the web map application and also gives some context of the work. The visualization approach will also be described in this chapter later on.

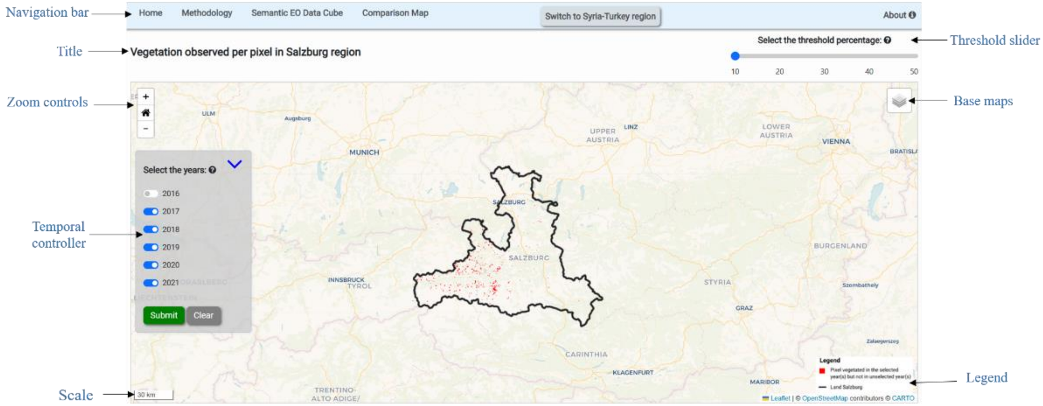


Fig 21: Home page of the application

b. **Title**: As the name suggests, the 'Title' represents concise heading about the information currently being represented on the map. In the application, the title of the map for the vegetation case study in Salzburg is '*Vegetation observed per pixel in Salzburg region*' and for the water occurrence case study is '*Water occurrence per pixel in Syria-Turkey border region for 2022'.*

c. **Threshold slider**: This component allows the users to select predefined threshold percentage. Users can move the slider to the desired value and that value will be selected automatically as a threshold percentage. Pixels with values above that threshold percentage will only be displayed on the map. For the vegetation case study, the thresholds were pre-defined as 10, 20, 30, 40 and 50 and for the water occurrence case study, they were established at values of 10, 20, 30, 40, 50, 60 and 70. As mentioned in chapter four, the range of percentage values in the data was taken into consideration to determine the predefined values for the threshold in the slider. There is also a symbol (question mark) which when hovered to, displays a brief explanation of the threshold percentage.

d. **Temporal controller:** Users can select one or multiple years from 2016 to 2021 in Salzburg or one or more months from January to December in Syria-Turkey by toggling on and off that particular timeframe. There are the 'Submit' and 'Clear' buttons, which on click, send requests to the server to add maps to the display and clear the display respectively. The *Temporal controller* and *Threshold slider* are the two key components that make the map interactive. Map users can manipulate the content displayed in the map by controlling these two components.

e. **Map:** Map is the most important component of the web application and covers the significant portion of the layout. The visualization in the map is controlled mainly by *Threshold slider* and *Temporal controller* components in the application. Users select one threshold percentage and one or more years or months depending on the study area. When the *Submit* button is clicked, all the pixels corresponding to that particular selection will be displayed in the map. The following map elements are present in the map and enhance the interactivity of it.

- Zoom controls
- Home button
- Scale
- Base maps selector
- Legend

Visualization approach in the map

The map displays pixels that were present in the 'selected years but not in unselected years'. So, only exclusive pixels present in the selected years will be displayed. The approach is illustrated with an example below.

Consider vegetation only for three years 2016, 2017 and 2018, as shown in the Venn diagram. If the user selects all three years 2016, 17 and 18, those pixels that were vegetated in all three years and the selected threshold percentage will be displayed. But if the user selects only 2016, only pixels vegetated in 2016 but not in 2017 and 2018 (red) will be displayed.

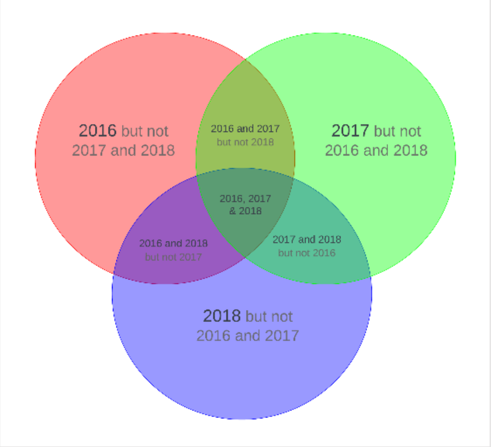

2016 but not 2017 and 2018
2016 and 2017 but not 2018
2017 but not 2016 and 2018
2016, 2017 & 2018
2016 and 2018 but not 2017
2017 and 2018 but not 2016
2018 but not 2016 and 2017

Fig 22: Illustration of visualization approach

The pixels that were vegetated commonly in 2016, 17 and 18 or in 2016 and 17 or in 2016 and 18 will not be displayed. So, three years are visualized in one layer. Though only three bands are shown in the Fig 17 for simplicity, the web application has visualization based on the same approach for six bands and 12 bands. Therefore, any kind of multiband data can be visualized in similar manner. It is similar to how RGB composites are created in GIS softwares where individual colours represent individual data and mix colours represent mixed data. But there is a limitation of representing complete information (6 years or 12 months of data like in this thesis) in similar manner in GIS softwares. Therefore, the web application attempts to visualize multi bands data treating them as one layer and using a single color.

Comparison map

There are two main maps, one for each use case in Salzburg and Syria-Turkey region. In addition to the main maps, there is also an opportunity for the users to compare two different data for the same study area with two maps side by side in both cases. The comparison map is expected to help the map users analyze the similarity, differences or any patterns in the data. The comparison maps also have similar map elements and layout like the main map. Users can select one set of time periods for the map on the left and another for the map on the right. The maps use two different colours in two maps to differentiate the data being represented. The maps on the left and right are synced to one another for zoom controls.

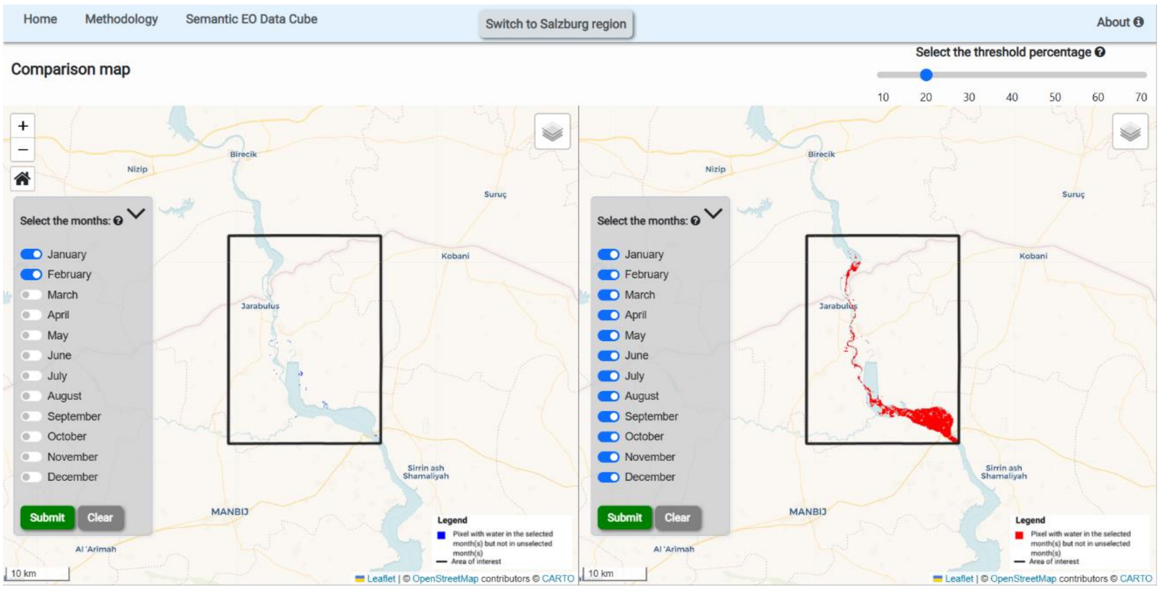


Fig 23: Interface for comparison map in Syria-Turkey region

6.2 Insights from the information

The number of unique pixels for each year was also calculated. In this work, unique pixels are considered to be those which were present only in that selected time period. These numbers give insights about vegetation dynamics of the region. The trend of vegetation growth or loss can be inferred. Vegetation is one of the most important elements of the ecosystem. The information about vegetation dynamics of any place can be used to understand the ecosystem process (Huo & Sun, 2021).

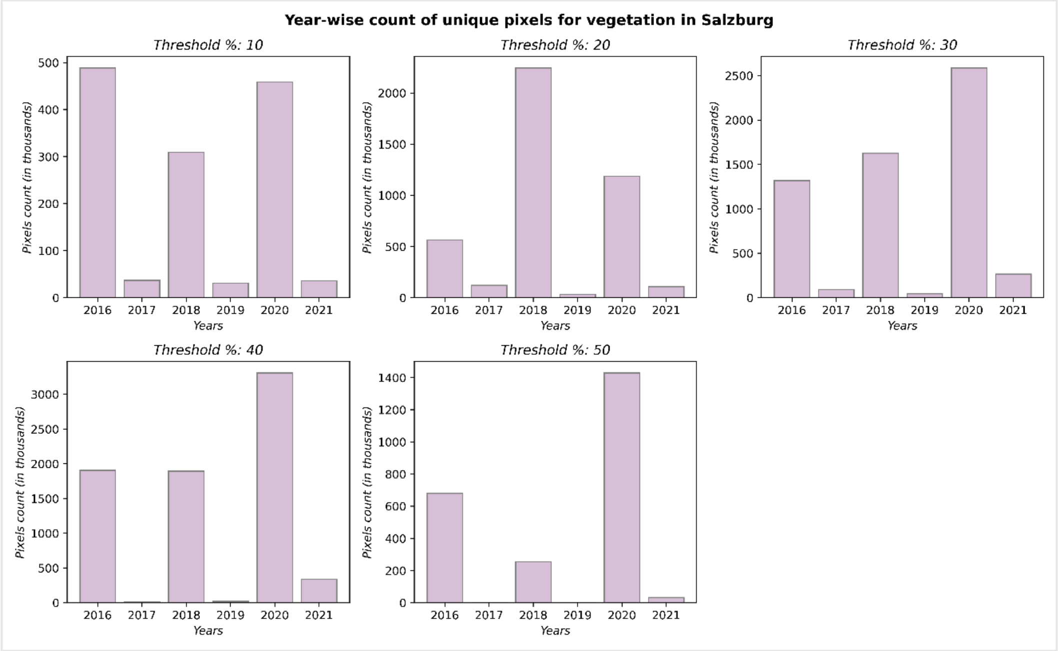


Fig 24: Unique pixels for different years and different threshold percentage

In the web application, when the user selects only 2021 and not the rest of the years for vegetation in Salzburg, the resulting pixels are the ones that appeared only in 2021. It means these pixels were not present in the previous five years. So, these pixels can be considered as vegetation gain in 2021. The number of pixels differs for different threshold percentage. In a similar manner, if the user selects all years from 2016 to 2020 but not 2021, the resulting pixels are those which were present in all four years but not in the last year. So, these pixels can be considered as vegetation loss in 2021. Each of the pixels has a 100 square meter area. Therefore, the vegetation loss and/or vegetation gain in 2021 can be expressed in terms of area as shown in Table 4.

Table 4 Vegetation gain and loss for different threshold percentage in 2021

Threshold %	Vegetation gain (km²)	Vegetation loss (km²)	Cumulative area (km²)	Gain/Loss
10	3.6299	59.894	56.2641	Loss
20	10.8897	42.7571	31.8674	Loss
30	26.6555	69.0865	42.431	Loss
40	33.9781	1.5586	-32.4195	Gain
50	3.3371	0	-3.3371	Gain

There is very less number of unique pixels for 2017, 2019 and 2021 for threshold percentages 40 and 50 as seen in Figure 24. The pattern is also not so different for the other three thresholds. The bar graph conveys the notion of less vegetation dynamics in the Salzburg region during these three years compared to the other three years 2016, 2018 and 2020. The study of the reasons for the change in vegetation dynamics is beyond the scope of the thesis.

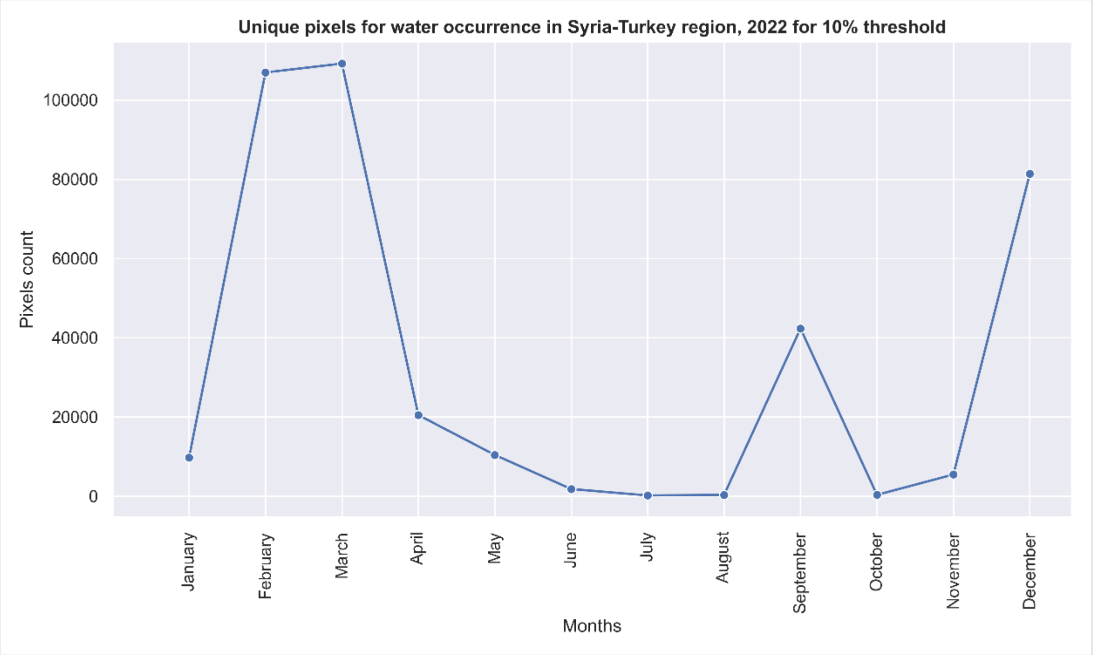


Fig 25: Unique pixels containing water in Syria-Turkey region with 10% threshold

The water occurrence in Syria-Turkey region was also projected to a graph. The finding states that the area contained more unique pixels during the months of February, March and December. There were very less unique pixels during the month of June, July and August, almost close to zero and there was a sudden spike in September. The full trend of the water dynamics can be perceived from the Figure 25. Similar patterns were observed with other threshold percentage.

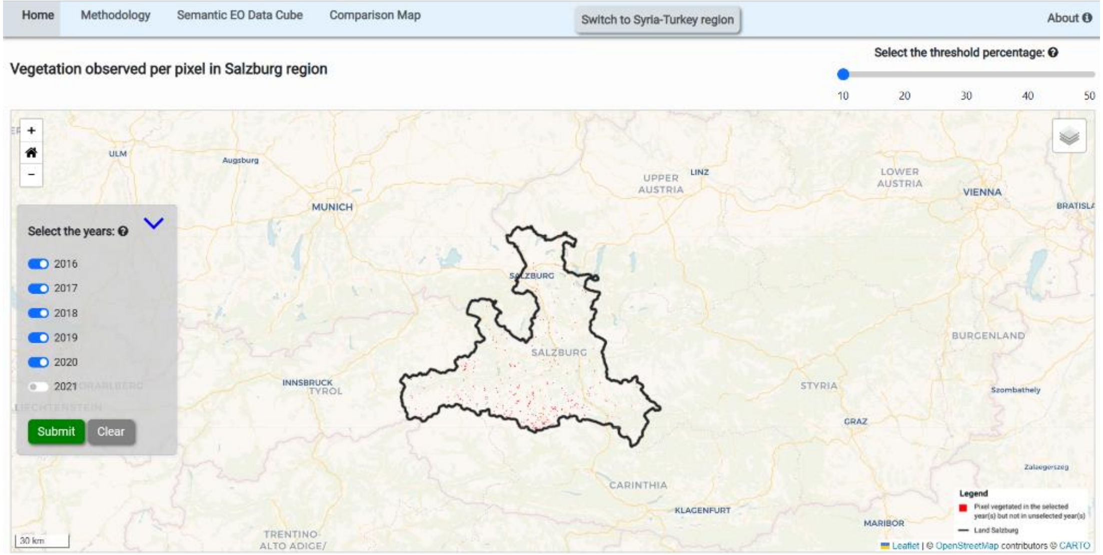
6.3 Findings from user evaluation

The web application was designed with a very specific purpose and its user group is very limited. The expected users are mostly people with some background in GIS, Remote Sensing and Cartography. Considering this, the survey was circulated to people with some technical background of these topics.

The goal was to understand if the information in the map can be interpreted by the map users as well as get the overall feedback on the cartographic and other components of the web application. The first part of the survey included questionnaires related to the user's understanding of the visualization approach used in the application. The second segment focused on the other components of the web application including colours, fonts and basemaps used. The last part was about understanding the respondent's background including their familiarity with the topics. The order of the questions was arranged in complex to simple hierarchy as mentioned in Chapter 5.

Sixteen participants responded to the survey out of which 10 were males and the rest of the others were female. The respondents were from the background of Geomatics Engineering, Geoinformatics, GIS, Geoscience, Geography, Geodesy, Environmental Science and Civil Engineering. Most of them had familiarity with Remote Sensing and related concepts. Two of the respondents never heard about earth observation data cubes before while the remaining others had some level of understanding about it. All of them except for one were in the age category of 21-24 and 25-30. The remaining one was in the age category of 31-40. All of the respondents had at least completed undergraduate studies.

The first six questions asked respondents to answer what red or blue color corresponds to in the map. Screenshots from the application were put in each question. About 65% of the responses were correct for these questions.



Vegetation observed per pixel in Salzburg region

Select the threshold percentage: 10

Select the years:
- ● 2016
- ● 2017
- ● 2018
- ● 2019
- ● 2020
- ○ 2021

Legend
- 🟥 Pixel vegetated in the selected year(s) but not in unselected year(s)
- — Land Salzburg

Fig 26: Pixels vegetated in 2016-2020 but not in 2021 with 10% threshold

About 65% of the responses were correct for these questions. Since the responses to these questions were text-based, the author considered those answers to be the correct ones where both the selected and unselected years/months were mentioned logically. For example: If the question was '*What does 'red' colour in the map below represents? What information is being shown? (Please be as specific as you can)*' in reference to Figure 26, the probable correct answers would be: i. *Vegetation observed in 2016, 2017, 2018, 2019, 2020 but not in 2021*; ii. *Vegetated pixels with 10% threshold from 2016-2020 but not in 2021*; or something similar. But if the respondents just answered '*vegetated pixels from 2016 to 2020'* or '*vegetated pixels from 2016, 2017, 2018, 2019, 2020 with 10% threshold*', that would be considered incorrect. Five points Likert scale was used to

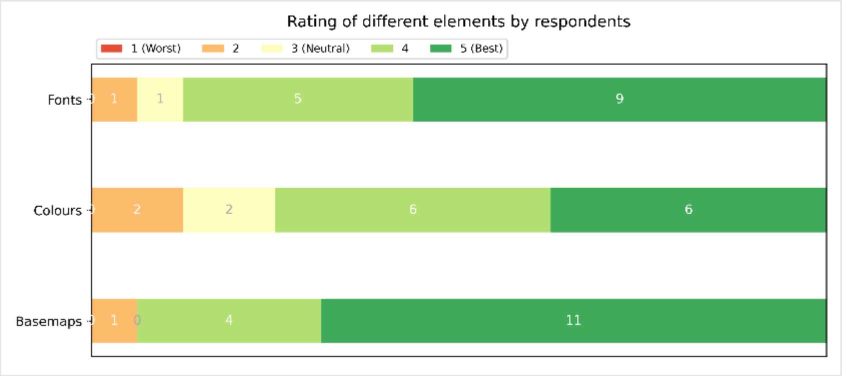


Fig 27: Perception of respondents towards fonts, colours and basemaps

enquire the user perception on fonts, colours and basemaps. Majority of the participants expressed positive perception towards all three elements**.** One of the participant was neutral about the fonts used and two of them were neutral about the colours. The full information is presented in Figure 27. When asked about what they liked about the application, the common answers were way of visualizing multi temporal information, side wise comparison, etc. But the respondents also felt the necessary of more comprehensive documentation about the approach with interesting examples, improvement in the interface and speed, etc.

7 DISCUSSION

Interactive web application with a focus on visualization of multi temporal information was created with the workflow devised as part of this thesis. In this chapter, the overall works done to achieve the objectives of the thesis are summarized. In addition to that, a critical evaluation is carried out on the entire process right from data, formulating the workflow, selection of technology stacks, pre-processing steps, development cycle, the product to the mechanism of user evaluation. The problems encountered during each phase will also be discussed.

Data and study area

The objective of the thesis was to visualize the information obtained from the semantic EO data cube, Sen2Cube.at in particular. As mentioned in Chapter 3, at the time of writing, Sen2Cube.at is implemented for the whole of Austria, some parts of Afghanistan and Syria and some regions in Europe. The selection of some regions of Austria and Syria as study areas was solely based on this fact. The accuracy of the visualization is highly dependent on the quality of multi-temporal information used. While in this case, the information obtained from Sen2Cube.at has the highest standard, consideration should be given to the credibility of the data when this approach is replicated with other data sources.

Selection of technology stacks

Broad range of tools were used to accomplish the objectives of the thesis. QGIS, Python, Geoserver, HTML, CSS and JS (Leaflet, jQuery) were some of them. The full list can be found in Figure 5. All of the tools used were free and/or open source. These tools were powerful enough to automate the processing steps and integrate the mapping and interactive elements to the web application.

Information pre-processing and hosting

The processing part was done manually in QGIS at first. Considering the volume of data, python scripts using PyQGIS were developed eventually and used to automate the routine tasks in QGIS yielding greater efficiency. This is particularly relevant when dealing with large volumes of data. Client to server connection was not established when rasters were stored in GeoServer hosted without encryption. So, another Geoserver encrypted with Secure Sockets Layer (SSL) was set up and data was hosted which solved the issue.

Web application

The interactive map is the primary component of the web application. The map contains features like zoom in, zoom out, zoom to home, base map change, etc. The spatio-temporal nature of the visualized information can be explored with the interactive elements present in the app. The visualization may not be intuitive to some users in the sense about how information is being shown. This problem was also discovered during user evaluation. For that purpose, the ‘About’ and ‘Methodology’ section briefly presents the approach of how information is being represented and the methodology behind it.

Small pop up box was added to some of the components (threshold slider and year/month selector) to assist users on their functions. The goal of adding all these components to the application was to assist users in getting better user experience and understand the message in the map. The author believes that incorporating additional visualizations like graphs or charts along with the main map can communicate the information more effectively. Bright colours like Red and Blue were used to represent pixels in the map because of their small size and to create visual dominance. This may not always be a right approach cartographically.

User evaluation

Survey based on Google forms was disseminated to people to test the usability of the application. 16 users participated in the evaluation. More user participation and thus increased responses provides more credibility to the results. It was also difficult to quantify some of the subjective answers of the survey. Remote methods like online surveys can be a quick way to check the usability of a product but quantitative methods have been proven to give more accurate results.

CONCLUSION

The volume of earth observation data being produced is increasing day by day. Extracting relevant information from the multitude of data is a challenge. Visualizations can play a big role in transforming data to information and eventually to knowledge (MacEachren & Kraak, 2001). Along that line, the thesis was developed to process the multi temporal information obtained from semantic earth observation data cube, develop workflow for visualizing that information and create visualizations based on the developed workflow. This document describes the aforementioned processes in detail.

The raster information obtained from the semantic data cube was pre processed in QGIS software using Python for automation. Subsequently, they were hosted in GeoServer remotely and an interactive web map application was developed to present the visualization on the web. The application was primarily developed with front-end technologies like HTML, CSS and JS along with their libraries and hosted on GitHub pages. The web application can be accessed at https://rabenojha.github.io/multi_temporal_infoviz/index.html. Two use cases of vegetation in Salzburg, Austria and water occurrence in northwestern Syria were demonstrated in the application. In addition to the web app, some other insights obtained from the data are also presented in this document. Lastly, an online survey was performed to test the usability of the web application. Some of the feedback were incorporated into the application.

Overall, the thesis utilizes the capacity of GIS software and modern web technologies to offer interactive ways to present multi temporal information to the users. Similar kind of multi temporal information from any sources can be visualized with the workflow developed in the thesis.

REFERENCES

Aigner, W., Miksch, S., Müller, W., Schumann, H., & Tominski, C. (2007). Visualizing time-oriented data—A systematic view. *Computers & Graphics*, *31*(3), 401–409. https://doi.org/10.1016/j.cag.2007.01.030

Andrienko, G., Demsar, U., Dransch, D., Dykes, J., Fabrikant, S., Jern, M., Kraak, M.-J., Schumann, H., & Tominski, C. (2010). Space, Time and Visual Analytics. *International Journal of Geographical Information Science*, *24*, 1577–1600. https://doi.org/10.1080/13658816.2010.508043

Augustin, H., Sudmanns, M., Tiede, D., Lang, S., & Baraldi, A. (2019). Semantic Earth Observation Data Cubes. *Data*, *4*(3), 102. https://doi.org/10.3390/data4030102

Ban, Y. (2016). *Multitemporal Remote Sensing: Current Status, Trends and Challenges* (pp. 1–18). https://doi.org/10.1007/978-3-319-47037-5_1

Bert, V., Brovelli, M., & Li, S. (2017). Review of Web Mapping: Eras, Trends and Directions. *ISPRS International Journal of Geo-Information*, *6*, 317. https://doi.org/10.3390/ijgi6100317

Bláha, J. (2009). *Various Ways of Assessment of Cartographic Works* (pp. 211–229). https://doi.org/10.1007/978-3-642-03294-3_14

Cartwright, W. (2003). Chapter 3—Maps on the Web. In M. Peterson (Ed.), *Maps and the Internet* (pp. 35–56). Elsevier Science. https://doi.org/10.1016/B978-008044201-3/50005-0

de la Beaujardiere, J. (2006). *OpenGIS® Web Map Server Implementation Specification. Version 1.3.0.* [Report]. Open Geospatial Consortium. https://doi.org/10.25607/OBP-656

Guo, J., Huang, C., & Hou, J. (2022). A Scalable Computing Resources System for Remote Sensing Big Data Processing Using GeoPySpark Based on Spark on K8s. *Remote Sensing*, *14*(3), Article 3. https://doi.org/10.3390/rs14030521

Harris, R., & Baumann, I. (2015). Open data policies and satellite Earth observation. *Space Policy*, *32*, 44–53. https://doi.org/10.1016/j.spacepol.2015.01.001

Huo, H., & Sun, C. (2021). Spatiotemporal variation and influencing factors of vegetation dynamics based on Geodetector: A case study of the northwestern Yunnan Plateau, China. *Ecological Indicators*, *130*, 108005. https://doi.org/10.1016/j.ecolind.2021.108005

Iacovella, S. (2014). *GeoServer Cookbook*. Packt Publishing Ltd.

Jianya, G., Guorui, M., & Zhou, Q. (2008). A review of multi-temporal remote sensing data change detection algorithms. *Remote Sensing and Spatial Information Sciences, Beijing, China, 3-11 Jul*, *37*.

Joshi, A., Kale, S., Chandel, S., & Pal, D. (2015). Likert Scale: Explored and Explained. *British Journal of Applied Science & Technology*, *7*(4), 396–403. https://doi.org/10.9734/BJAST/2015/14975

Kalbandi, I., & Anuradha, J. (2015). A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology. *Procedia Computer Science*, *48*, 319–324. https://doi.org/10.1016/j.procs.2015.04.188

Killough, B. (2018). *Overview of the Open Data Cube Initiative* (p. 8632). https://doi.org/10.1109/IGARSS.2018.8517694

Kraak, J.-M., & Brown, A. (2001). *Web Cartography*.

Kshetri, T. B., Chaksan, A., & Sharma, S. (2021). THE ROLE OF OPEN-SOURCE PYTHON PACKAGE GEOSERVER-REST IN WEB-GIS DEVELOPMENT. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLVI-4-W2-2021*, 91–96. https://doi.org/10.5194/isprs-archives-XLVI-4-W2-2021-91-2021

Ma, Y., Wu, H., Wang, L., Huang, B., Ranjan, R., Zomaya, A., & Jie, W. (2015). Remote sensing big data computing: Challenges and opportunities. *Future Generation Computer Systems*, *51*, 47–60. https://doi.org/10.1016/j.future.2014.10.029

MacEachren, A. M., & Kraak, M.-J. (2001). Research Challenges in Geovisualization. *Cartography and Geographic Information Science*, *28*(1), 3–12. https://doi.org/10.1559/152304001782173970

Muehlenhaus, I. (2013). *Web Cartography: Map Design for Interactive and Mobile Devices*. CRC Press. https://doi.org/10.1201/b16229

NASA - NSSDCA - Master Catalog—Spacecraft Query. (n.d.). Retrieved May 13, 2023, from https://nssdc.gsfc.nasa.gov/nmc/spacecraft/query

Nativi, S., Mazzetti, P., Santoro, M., Papeschi, F., Craglia, M., & Ochiai, O. (2015). Big Data challenges in building the Global Earth Observation System of Systems. *Environmental Modelling & Software*, *68*, 1–26. https://doi.org/10.1016/j.envsoft.2015.01.017

Neumann, A. (2008). Web Mapping and Web Cartography. In S. Shekhar & H. Xiong (Eds.), *Encyclopedia of GIS* (pp. 1261–1269). Springer US. https://doi.org/10.1007/978-0-387-35973-1_1485

Peuquet, D. J. (1994). It’s about Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. *Annals of the Association of American Geographers*, *84*(3), 441–461.

Popelka, S., Herman, L., Řezník, T., Pařilová, M., Jedlička, K., Bouchal, J., Kepka, M., & Charvat, K. (2019). User Evaluation of Map-Based Visual Analytic Tools. *International Journal of Geo-Information*, *8*. https://doi.org/10.3390/ijgi8080363

Rizvi, S. R., Killough, B., Cherry, A., & Gowda, S. (2018). The Ceos Data Cube Portal: A User-Friendly, Open Source Software Solution for the Distribution, Exploration, Analysis, and Visualization of Analysis Ready Data. *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 8639–8642. https://doi.org/10.1109/IGARSS.2018.8518727

Rono, D. (2020, December 10). Harness the power of Sentinel Hub, xcube, EOxHub, GeoDB and more in Euro Data Cube. *Euro Data Cube*. https://medium.com/euro-data-cube/harness-the-power-of-sentinel-hub-xcube-eoxhub-geodb-and-more-in-euro-data-cube-31c827a3083b

Roth, R. (2011). *Interacting With Maps: The Science And Practice Of Cartographic Interaction*.

Roth, R. E. (2012). Cartographic Interaction Primitives: Framework and Synthesis. *The Cartographic Journal*, *49*(4), 376–395. https://doi.org/10.1179/1743277412Y.0000000019

Semantic Earth Observation Data Cube Analysis. (2023, January 30). ArcGIS StoryMaps. https://storymaps.arcgis.com/stories/05c3980c712b4190b54f7ae3156ae61e

Sentinel-2—Missions—Sentinel Online. (n.d.). Sentinel Online. Retrieved May 20, 2023, from https://copernicus.eu/missions/sentinel-2

Simoes, R., Camara, G., Queiroz, G., Souza, F., Andrade, P. R., Santos, L., Carvalho, A., & Ferreira, K. (2021). Satellite Image Time Series Analysis for Big Earth Observation Data. *Remote Sensing*, *13*(13), Article 13. https://doi.org/10.3390/rs13132428

Sterba, Z., Šašinka, Č., & Konečný, M. (2014). *Usability Testing of Cartographic Visualizations: Principles and Research Design*.

Strobl, P., Baumann, P., Lewis, A., Szantoi, Z., Killough, B., Purss, M., Craglia, M., Nativi, S., Held, A., & Dhu, T. (2017). *The Six Faces of the Data Cube*.

Styled Layer Descriptor. (n.d.). *Open Geospatial Consortium*. Retrieved May 9, 2023, from https://www.ogc.org/standard/sld/

Sudmanns, M., Augustin, H., Killough, B., Giuliani, G., Tiede, D., Leith, A., Yuan, F., & Lewis, A. (2022). Think global, cube local: An Earth Observation Data Cube's contribution to the Digital Earth vision. *Big Earth Data*, *0*(0), 1–29. https://doi.org/10.1080/20964471.2022.2099236

Sudmanns, M., Augustin, H., van der Meer, L., Baraldi, A., & Tiede, D. (2021). The Austrian Semantic EO Data Cube Infrastructure. *Remote Sensing*, *13*(23), Article 23. https://doi.org/10.3390/rs13234807

Sudmanns, M., Tiede, D., Lang, S., Bergstedt, H., Trost, G., Augustin, H., Baraldi, A., & Blaschke, T. (2020). Big Earth data: Disruptive changes in Earth observation data management and analysis? *International Journal of Digital Earth*, *13*(7), 832–850. https://doi.org/10.1080/17538947.2019.1585976

Tiede, D., Sudmanns, M., Augustin, H., Lang, S., & Baraldi, A. (2019). *SENTINEL-2 SEMANTIC DATA & INFORMATION CUBE AUSTRIA*.

van der Meer, L., Sudmanns, M., Augustin, H., Baraldi, A., & Tiede, D. (2022). SEMANTIC QUERYING IN EARTH OBSERVATION DATA CUBES. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLVIII-4-W1-2022*, 503–510. https://doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-503-2022

WMS output formats—GeoServer 2.24.x User Manual. (n.d.). Retrieved May 11, 2023, from https://docs.geoserver.org/latest/en/user/services/wms/outputformats.html

Wu, X., Chen, Z., Gu, Y., Chen, W., & Fang, M. (2018). Illustrative visualization of time-varying features in spatio-temporal data. *Journal of Visual Languages & Computing*, *48*, 157–168. https://doi.org/10.1016/j.jvlc.2018.08.010

ATTACHMENTS

1. Combined script for raster thresholding and raster addition

```
# This script should be run in QGIS Python Console. To run it otherwise,
you need to set up the QGIS environment first.
# The script reads all the raster layers in the current QGIS project,
and creates a new binary raster layer for each input raster layer and
then create a final output raster layer by adding all the binary raster
layers together.

# import necessary libraries
from qgis.analysis import QgsRasterCalculator, QgsRasterCalculatorEntry
from qgis.core import QgsProject, QgsRasterLayer

# set the threshold value
threshold = 20

# get a list of all raster layers in the current QGIS project
layers = list(QgsProject.instance().mapLayers().values())
layers.sort(key=lambda x: int(x.name().split('_')[1]))

# initialize a list to hold the output layers
output_layers = []

for i, layer in enumerate(layers):
    # check if the layer is a raster layer
    if isinstance(layer, QgsRasterLayer):
        # assign the multiplication factor based on the layer order
        factor = 2 ** (len(layers) - i)

        entries = []
        ras = QgsRasterCalculatorEntry()
        ras = QgsRasterCalculatorEntry()
        ras.ref = layer.name() + '@1'
        ras.raster = layer
        ras.bandNumber = 1
        entries.append(ras)
        formula = f'(({ras.ref} > {threshold}) * {factor})'
        # set the output path
        output_path =
f'path_to_output_directory/output/{layer.name()}_{threshold}_binary.tif'
     calc = QgsRasterCalculator(formula, output_path, 'GTiff',
layer.extent(), layer.width(), layer.height(), entries)
        calc.processCalculation()
        # add the output layer to the output_layers list
        output_layer = QgsRasterLayer(output_path,
f'{layer.name()}_{threshold}_{factor}_binary')
        output_layers.append(output_layer)

print('halfway!!')

# set up the raster calculator to add the output layers together
entries = []
for layer in output_layers:
    ras = QgsRasterCalculatorEntry()
    ras.ref = layer.name() + '@1'
    ras.raster = layer
    ras.bandNumber = 1
    entries.append(ras)
formula = ' + '.join([f'({ras.ref})' for ras in entries])
# set the output path
output_path =
'path_to_output_directory/output/final_output{}.tif'.format(threshold)
calc = QgsRasterCalculator(formula, output_path, 'GTiff',
layers[0].extent(), layers[0].width(), layers[0].height(), entries)
calc.processCalculation()

# add the final output layer to the layer panel
final_output_layer = QgsRasterLayer(output_path,
f'final_output_{threshold}')
QgsProject.instance().addMapLayer(final_output_layer)
print('The process is completed')
```

LIST OF ATTACHMENTS

Bound attachments

Attachment 1 Python script for raster thresholding and addition step

Free attachments

Attachment 2 Poster

Attachment 3 Flash drive

Flash drive file structure

Root directory	Sub directory
poster	
python_scripts	raster_thresholding_and_addition.py
	rasters_per_unique_value.py
	generate_sld.py
	upload_files_to_GeoServer.ipynb
survey_questionnaires	
survey_answers	
thesis_text	
web_application_code	
website	