ISPRS Journal of Photogrammetry and Remote Sensing 228 (2025) 552–565

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

On-demand, semantic EO data cubes – knowledge-based, semantic querying of multimodal data for mesoscale analyses anywhere on Earth

Felix Kröber [a,b,*], Martin Sudmanns [a], Lorena Abad [a,1], Dirk Tiede [a,*]

[a] Paris-Lodron University Salzburg, Department of Geoinformatics - Z_GIS, Schillerstraße 30, 5020 Salzburg, Austria
[b] Research Centre Jülich, Institute of Bio- and Geosciences, Leo-Brandt-Straße, 52425 Jülich, Germany

ARTICLE INFO

Dataset link: https://github.com/Sen2Cube-at/gsemantique, https://doi.org/10.5281/zenodo.15423258

Keywords:
Earth observation
Remote sensing
Big data analyses
Data cubes
Semantic querying

ABSTRACT

With the daily increasing amount of available Earth Observation (EO) data, the importance of processing frameworks that allow users to focus on the actual analysis of the data instead of the technical and conceptual complexity of data access and integration is growing. In this context, we present a Python-based implementation of ad-hoc data cubes to perform big EO data analysis in a few lines of code. In contrast to existing data cube frameworks, our semantic, knowledge-based approach enables data to be processed beyond its simple numerical representation, with structured integration and communication of expert knowledge from the relevant domains. The technical foundations for this are threefold: Firstly, on-demand fetching of data in cloud-optimized formats via SpatioTemporal Asset Catalog (STAC) standardized metadata to regularized three-dimensional data cubes. Secondly, provision of a semantic language along with an analysis structure that enables to address data and create knowledge-based models. And thirdly, chunking and parallelization mechanisms to execute the created models in a scalable and efficient manner. From the user's point of view, big EO data archives can be analyzed both on local, commercially available devices and on cloud-based processing infrastructures without being tied to a specific platform. Visualization options for models enable effective exchange with end users and domain experts regarding the design of analyses. The concrete benefits of the presented framework are demonstrated using two application examples relevant for environmental monitoring: querying cloud-free data and analyzing the extent of forest disturbance areas.

1. Introduction

In recent years, Earth Observation (EO) data access and processing has changed in a fundamental way. The opening of the Landsat archive in 2008 (Woodcock et al., 2008) enabled broad-scale analyses driven by a steadily growing user base (Wulder et al., 2012; Zhu et al., 2019). Free access to global satellite data at unprecedented spatial and temporal resolution has been further advanced by the Sentinel-2 (S-2) constellation (Drusch et al., 2012) launched in 2015. To facilitate the exploitation of the data, a new paradigm of big EO data processing has emerged (Guo et al., 2017; Sudmanns et al., 2020b). Among the underlying factors powering the evolution of this field, there are two major ones that keep posing challenges to users.

Firstly, accessible EO data is continuously increasing in terms of their volume. The Copernicus Open Access Hub, meanwhile replaced by the Copernicus Data Space Ecosystem, for example, provided a total of over 45 petabyte of data, which had been continuously built up over the previous years (Cipoletta and Sciarra, 2024). While this data availability enables broad-scale analyses in space and time, in practice there are usually limitations on users' ability to process large amounts of data. In addition to specific approaches such as optimizing the data selection (Kempeneers and Soille, 2017), these limitations gave rise to a fundamental change in data management. Instead of downloading data and processing it locally, throughout the last 10 years it became more common to analyze data in the cloud utilizing data models including data cubes (Sudmanns et al., 2020b). Still, a majority of users report limiting processing capabilities and growing data volumes as prevailing obstacles when working with big EO data (Wagemann et al., 2021). One reason for this can be seen in a reluctance to fully shift to cloud-based processing platforms due to their ongoing limitations. Many of these platforms are proprietary, closed-source (Gorelick et al., 2017; Microsoft Open Source et al., 2022), and their usage usually incurs costs. The lack of guarantees on the provision of the service can lead to

* Corresponding authors at: Paris-Lodron University Salzburg, Department of Geoinformatics - Z_GIS, Schillerstraße 30, 5020 Salzburg, Austria.
E-mail addresses: felix.kroeber@plus.ac.at (F. Kröber), martin.sudmanns@plus.ac.at (M. Sudmanns), lorena.abad@plus.ac.at (L. Abad), dirk.tiede@plus.ac.at (D. Tiede).
[1] Deceased author.

https://doi.org/10.1016/j.isprsjprs.2025.07.015
Received 13 February 2025; Received in revised form 20 May 2025; Accepted 8 July 2025
Available online 29 July 2025

unexpected shutdowns impeding analysis reproducibility, most recently experienced in June 2024 with Microsoft Planetary Computer Hub. Furthermore, such platforms exhibit restricted flexibility in terms of extensibility and customization. Meanwhile, open-source alternatives for large-scale data processing exist, but they are usually tedious to set up, e.g. due to data set indexing (Killough, 2018; Baumann et al., 2018). This effort currently limits the usability of open-source frameworks, especially for projects with a shorter runtime.

Secondly, supported by the launch of new satellites and sensors, the variety of data sets keeps growing. A rise in the availability of data sets with different spectral, spatial, temporal and radiometric resolutions poses challenges for multimodal data sources analyses. The SpatioTemporal Asset Catalog (STAC) fostering standardization in the structuring and publishing of geospatial metadata is an important means to facilitate data access. But data utilization and integration are not only about the technical means to query data. Data access is a natural prerequisite but by itself it does not support sophisticated image analytics. An example for optical images is the image understanding process of inferring information on 4-D physical world phenomena using a numerical model that runs on the 2D image domain. In order for EO data analysts to focus on image understanding they need to be provided with adequate means, allowing to query information and model knowledge in a consistent and transparent manner. Querying frameworks that provide a structured approach for EO information extraction are recently evolving (Van Der Meer et al., 2022) but not yet supported in most of the existing EO data cube systems. Many of them focus on technical solutions in terms of provision of data but they do lack the building blocks to aid knowledge-based image understanding.

In brief, we acknowledge the pressing need for open-source big EO data processing frameworks, which are easy-to-use, while having a sufficient conceptual basis to be able to deal with the semantics of EO data. To tackle this gap, we propose a data cube approach based on a semantic querying language that interprets knowledge embedded in semantic models to support big EO operational image understanding. Specifically, the contribution of this paper is a new Python package *gsemantique* that enables building ad hoc data cubes for semantic, knowledge-based EO analyses in on-premise or cloud-based infrastructures. We demonstrate the potential value of this implementation with two use cases.

This paper is structured as follows. In Section 2, we place our package in relation to existing data cube frameworks and recapitulate on the essence of semantic, knowledge-based querying. In Section 3, the technical implementation of the package and underlying design choices are presented. With Section 4 we showcase the general usage of the package. This is followed by the presentation of two specific use cases in Section 5, one focused on cloud-free imagery and the other one on forest disturbances. The paper concludes with a reflection on limitations and future works in Section 6 and a summary in Section 7.

2. Related works & conceptual foundations

2.1. EO data cubes

A data cube is a multi-dimensional array whose grid points are populated with data of the same data type (Baumann, 2017). The data values are indexed unequivocally by coordinates along the d axes of the d-dimensional data cube. In the EO domain, data cubes typically have at least two spatial and one temporal dimension, and the coordinates span the full spatiotemporal extent of a given set of observations (Lewis et al., 2016). The primary feature of EO data cubes is that the data is reorganized such that from a logical view data can be queried easily using spatiotemporal coordinates and abstraction of analytics from storage considerations. This data organization replaces the traditional file-based access for users, which is limited by files being organized in nested directory structures in a spatiotemporally inconsistent manner with custom naming patterns. EO data cubes therefore provide more convenient access to data, facilitating their analyses by reducing the pre-processing effort, which in turn is closely linked to the provision of analysis ready data (Giuliani et al., 2017). Beyond its array structure, conceptual disagreement about the essence of a data cube is still prevalent. Baumann et al. (2016) defined a set of technical requirements data cubes should adhere to. Strobl et al. (2017) extended this set of properties by providing a holistic view on six system-level aspects that need to be considered to realize the full potential of data cubes. Despite the valuable criteria provided, practical implementation considerations result in a broad variety of systems currently operating under the term ‘EO data cube‘. Therefore, we stick with the universal array definition of the EO data cube and subsequently highlight the specifics of our approach by comparing it with other EO data cube implementations. Note that proprietary geospatial web-based processing platforms including Google Earth Engine (Gorelick et al., 2017) and Microsoft Planetary Computer (Microsoft Open Source et al., 2022) are deliberately excluded from the comparison. A comprehensive overview on web-based processing frameworks can be found in Gomes et al. (2020). For the difference between such infrastructures and EO data cubes, the reader is referred to Giuliani et al. (2019).

One of the first operational national-wide EO data cube implementations was the Australian Geoscience Data Cube (AGDC) (Lewis et al., 2016). Whereas initially data was ingested, i.e. restructured via resampling and tiling, further developments shifted towards data indexing (Lewis et al., 2017), where the data is stored in its native format without being replicated. The evolution of the AGDC gave rise to the Open Data Cube (ODC) initiative (Killough, 2018) providing a set of open-source tools to create data cube infrastructures. The ODC approach gained attention rapidly (Dhu et al., 2019; Killough et al., 2020) with a variety of data cubes representing ODC instances including, for example, the Swiss Data Cube (Giuliani et al., 2017; Chatenoux et al., 2021), the Colombian Data Cube (Ariza-Porras et al., 2017), the Armenian Data Cube (Asmaryan et al., 2019), the Catalan Data Cube (Maso et al., 2019), the Vietnam Open Data Cube (Quang et al., 2019), the Austrian Semantic EO Data Cube (Sudmanns et al., 2021) and Digital Earth Africa (Yuan et al., 2021). An alternative array data base solution is provided by Rasdaman (Baumann et al., 2018) deployed in Baumann et al. (2016) and Storch et al. (2019), where the data is tiled into sub-arrays according to specific partitioning strategies and ingested into a data base to optimize the retrieval efficiency. All of the above-mentioned systems are united by their property of being extensive software infrastructures consisting of several components (e.g. modules for data pre-processing, data bases, APIs for data querying, monitoring tools).

Some efforts have been made to lower the hurdles for setting up such complex software infrastructures. In line with the idea of self-hosted deployments of local, federated EO data cubes (Sudmanns et al., 2023), Giuliani et al. (2020) proposed a proof-of-concept for the automated generation of ODC instances. The user only needs to specify an area, time frame and sensor of interest to retrieve an ODC instance. As an all-in-one solution, Frantz (2019) proposed FORCE for the processing of large amounts of S-2 and Landsat data. Without any data base-driven indexing or ingestion, FORCE provides a suite of algorithms to create regularly tiled, analysis ready data on Level 2 or even higher levels, where all data for a given tile is referred to as a data cube. Despite these developments, from the user’s point of view the time required for the initial creation of a populated data cube is quite high as, in the cases mentioned, data cubes are created with the original data being first downloaded and persisted on the disc. This results in static, infrastructure-oriented data cubes that are tailored to a few data products. In contrast, nowadays, many usable EO data products are already available on the web as analysis ready data, and a fast and flexible integration of different data products for on-the-fly analyses is desired.

The idea of on-demand or on-the-fly cubes summarizes approaches to create data cubes in an ad-hoc fashion for any specified spatiotemporal extent of interest. While lacking some of the functionalities and

performance benefits of more comprehensive data cube approaches, the on-demand approaches have the advantages of being lightweight and easy-to-use. xcube (Brockmann Consult GmbH, 2021) creates self-contained EO data cubes by relying on Python's big data ecosystem, specifically xarray (Hoyer and Hamman, 2017) for in-memory representations, dask (Dask Development Team, 2016) for memory management and Zarr as a format for cloud-native, chunked storage. Implemented in the Euro Data Cube (Euro Data Cube Consortium) and the multivariate Earth system data cubes as part of the Earth System Data Lab project (Mahecha et al., 2020), xcube is used in operational systems. Supplemented by the ml4xcube library (Peters et al., 2025), not only the creation but also the data-driven analysis of EO data cubes is facilitated. To automate the creation of mini data cubes as xarray objects from STAC catalogs, cubo has been proposed by Montero et al. (2024a). The open source C++ library gdalcubes (Appel and Pebesma, 2019) natively provides chunked management and parallel processing of EO data cubes. It can integrate with scripting languages such as R, Python or Julia or with software that can handle data cubes such as GRASS GIS (Neteler et al., 2012). Its R implementation depends on the stars package (Pebesma and Bivand, 2023), which enables the reading and processing of spatiotemporal arrays and allows proxy objects with lazy loading for larger rasters. Additionally, gdalcubes offers a set of predefined formats to load various EO products as image collections and convert them to regularized data cubes. Image collections can also be built from STAC catalogs assets accessed via the rstac package (Simoes et al., 2021b), developed as part of the Brazil Data Cube (BDC) project (Ferreira et al., 2020). A more comprehensive effort, also stemming from the BDC project, is the sits package (Simoes et al., 2021a), built on top of gdalcubes. sits has a tailored focus on satellite image time series analysis using data-driven techniques. It allows to build data cubes from various cloud-based providers of EO images and train machine and deep learning models on them with support for parallel processing and chunking along the spatial dimension. While most of the listed approaches offer a specific solution for data cube creation, only some offer end-to-end frameworks that allow to realize a full processing pipeline with the final aim of producing tailored information from EO data. The framework proposed by us supports end-to-end EO analysis with a specific focus on semantic querying and knowledge integration. The essence and relevance of semantic, knowledge-based querying for remote sensing-based image understanding is detailed further in Section 2.2.

2.2. Remote sensing based image understanding

The general vision process based on remote sensing data amounts to reconstructing a semantic 4D scene reality from sub-symbolic 2D image data (Matsuyama and Hwang, 1990). This makes it an inherently ill-posed problem. The following describes how semantic and knowledge-based systems, which are forming the basis for the analysis backbone of our on-demand EO cubes, deal with this complex task.

2.2.1. Semantic systems

Semantics, as the study of meaning, deals with the relation between physical world phenomena, mental concepts, and the expressions used to interconnect both. Enabling machines to be capable of handling semantics is fundamental for any advanced human-machine or machine-machine interaction, and not specific to the EO domain, as exemplified by the semantic web (Berners-Lee et al., 2001). A central feature of semantically-enabled systems is that beyond data itself, information as interpretations of data can be accessed and handled. In the EO context, semantic data cubes thus refer to systems that leverage interpretations of EO images, where for each spatiotemporal observation at least one interpretation is available (Augustin et al., 2019). Those interpretations are effectively mapping numerical sensory data to stable concepts. Semi-symbolic spectral categorizations as proposed by Baraldi (2011), for example, provide low level interpretations. They allow an initial characterization of the data, e.g. by splitting the continuous multispectral reflectance space into a discrete set of physically meaningful categories, but they do not represent physical world entities. In contrast, high level interpretations are given by concepts that adhere to existing ontologies describing physical world entities such as land cover classes according to the FAO LCCS (Di Gregorio et al., 2016). The common property of both types of interpretations is that they represent semantic enrichments transforming continuous, numeric data into interpretable categorical data, which essentially lifts elements from the lower data level to the next level of the data-information-knowledge-wisdom hierarchy (Rowley, 2007). Since the essence of image analysis and understanding is to transform data into information, most computer vision tasks including EO analyses are inherently dealing with questions of semantics.

2.2.2. Knowledge-based systems

Knowledge is a vague concept, but commonly referred to as structured, contextualized, and synthesized information (Rowley, 2007). Relevant for computational systems, knowledge enables to translate information into instructions, thereby allowing the guidance of systems (Ackoff, 1989). For the purposes of an image understanding system, various types of knowledge are required. Those include generic knowledge on problem solving and image understanding, remote sensing domain specific knowledge to achieve a meaningful mapping between numeric and symbolic representations, and knowledge on user interfaces to allow interaction with humans as knowledgeable, intelligent system users (Crevier and Lepage, 1997). There is a variety of knowledge representation techniques dealing with how knowledge is embedded and represented in systems. In the field of image understanding, those representation techniques are essentially aiming to transform data into information, i.e. to gain actionable insights from data. Baltsavias (2004) presents an overview of knowledge representations that are commonly used in the domain of remote sensing-based image understanding. They belong to the realm of more established, old-school artificial intelligence systems.

More recently, the field of image understanding has been supplemented and in large parts dominated by the suite of machine and deep learning techniques (Mountrakis et al., 2011; Belgiu and Drăguţ, 2016; Zhu et al., 2017; Hoeser and Kuenzer, 2020; Hoeser et al., 2020). While the design of these techniques and their selection for a specific task certainly involve knowledge, the actual reasoning process itself is carried out in an inductive, data-driven way. In the classical supervised paradigm, machine and deep learning techniques are essentially statistical models learning from examples. Baraldi and Boschetti (2012) and Baraldi et al. (2023) argue that these models are not only poorly based on correlation instead of causation, but that the image understanding process with these models remains ill-posed, because external input is required for the scene reconstruction. They emphasize the need for a-priori knowledge to make the vision problem better conditioned for solution. While data-driven models can be adapted by incorporating a-priori knowledge, one can argue that knowledge-based systems operating in a deductive manner remain a valuable complementary approach to tackle image understanding. Arvor et al. (2021) promote knowledge-based approaches focusing on their advantage to explicitly deal with symbolic information and its organization. Craglia and Nativi (2018) emphasize that specifically in the big data era with the prevalence of data-driven inferences, a focus on transparent modeling and exchange of domain knowledge is needed to increase the trust in inferences made on big data. Scheider et al. (2017) focus on the relevance of integrating existing knowledge into analyses for reproducible generation of information and further knowledge instead of pursuing knowledge acquisition in a purely data-driven manner. This coincides with a broader epistomological belief that despite the prevalence of data-driven approaches, the synergy of deductive and inductive approaches is required to drive information derivation and knowledge acquisition (Mazzocchi, 2015).

With our system design, we are referring to expert systems (Good-enough et al., 1987; Laurini and Thompson, 1992; Matsuyama, 1993), which are representing the knowledge of a human expert in a declarative manner. Declarative knowledge ("knowledge-that") refers to factual information regarding physical world entities and their properties, and is often contrasted with practical knowledge ("knowledge-how") and knowledge by acquaintance ("knowledge-of"). Declarative knowledge can be represented explicitly, e.g. as symbolic data embedded in logical production rules such as "if-then" constructs. Importantly, the knowledge is independent from the procedural process of reasoning and its usage is not tied to a single use case. In terms of system design, this explains the typical decomposition of an expert system into two parts: the knowledgebase as a collection of symbolic data and the reasoning engine as the task-solving program that infers information by leveraging the knowledgebase within a specific reasoning process. The separation of knowledge from reasoning fosters modularity and enables knowledge sharing.

2.2.3. Synthesis: Semantic, knowledge-based systems

Given the descriptions on semantically-enabled and knowledge-based systems, one may ask to what extent both approaches can be seen independent of each other. Are all semantically-enabled systems inevitably knowledge-based systems and vice versa? Semantic enrichments are not exclusively generated by knowledge-based methods but can also be produced by data-driven methods (Baraldi and Boschetti, 2012). Semantically-enabled systems can be used within knowledge-based expert systems but can also be described on their own (Augustin et al., 2019). Knowledge-based systems aiming at image understanding are always concerned with semantics but not necessarily in an explicit manner. The degree of formalization of shared conceptualizations known as ontologies can be low in common rule-based modeling approaches that leverage symbolic knowledge implicitly (Arvor et al., 2019). Furthermore, knowledge-based systems do not per se target the inclusion of semantically enriched information in their reasoning processes. Therefore, semantic and knowledge-based systems are neither synonymous nor dependent on each other, but can indeed be used in a complementary, synergistic manner.

As a foundational work for the design of a corresponding semantic, knowledge-based image understanding system, we refer to the Austrian semantic data cube (Sudmanns et al., 2021). Following the design of an expert-based system, semantic models from the knowledgebase are processed by an inference engine against the factbase containing image data. Following the idea of semantically-enabled systems, the factbase stores additional semantically enriched information layers, and the inference engine allows to create and execute semantic models that are targeted to processing such categorical data. The way knowledge can be represented and embedded in semantic models is formalized via a semantic querying language, *semantique*, as described by Van Der Meer et al. (2022). Similar to Sudmanns et al. (2021), with our proposed *gsemantique* package, we built on top of this querying language to extend it towards an operational image understanding system. In contrast to Sudmanns et al. (2021) and according to Section 2.1, our system focuses on the creation of on-demand cubes in an ad-hoc fashion. To further illustrate the relationship between the existing systems in the field of semantic, knowledge-based image understanding, the reader is referred to Fig. 1.

3. System requirements & architecture

3.1. Design goals

We define the following design goals for our data cube framework. The goals are framed as user's requirements reflecting the specific expectations of users interacting with an on-demand data cube system that aims to support ad-hoc EO analyses and image understanding.

(A) Data Access

- (A.1) Coverage: Users can query data world-wide for any spatial or temporal extent of interest having a range of predefined data sets at hand
- (A.2) Extensibility: Users can easily index new data sets to integrate them in their analyses
- (A.3) Persistence: Data for generating the data cube can be persisted locally or in the cloud to enable inspection of input data, ensure reproducibility of analyses, and speed up the calculations in case of repeated execution of similar analyses by moving the data location closer to the computing infrastructure

(B) Analysis

- (B.1) Basic support: Provision of a standard set of spatial and temporal data cube operations
- (B.2) Semantic support: Modeling image semantics including…
  - model formulation from a semantic point of view and automated translation into procedural data cube operations
  - possibilities for custom modeling of entities of interest
  - support for categorical data operations to interact with semantically enriched data
- (B.3) Expert knowledge support: Means to represent expert knowledge in a model
- (B.4) Customization: Possibility to define analysis workflows including user-defined functions
- (B.5) Visualization, Communication, Exchange: Automated export and visualization of models to exchange with other domain experts and communicate models

(C) Processing

- (C.1) Scalability: Support for analyses with spatiotemporal extents up to mesoscale dimensionality
- (C.2) Efficiency: Fast data access and processing, minimization of redundancies in data loading, and exploitations of available processing resources
- (C.3) Abstraction of complexity: Automated model execution requiring minimal user interaction

(D) General software requirements

- (D.1) Portability: Executability on local devices as well as cloud-based platforms
- (D.2) Usability: Simple client-side installations via package managers; usability via common Python programming language enabling big EO analysis in a few lines of code

3.2. Implementation

To implement these requirements, we extended the existing semantic querying language *semantique* (Van Der Meer et al., 2022) and built a new package, *gsemantique*, on top of it. In terms of functionality, *semantique* represents the general modeling framework and inference engine responsible for the core analysis support (requirements (B)). *gsemantique* represents a wrapper around *semantique* to ensure data access

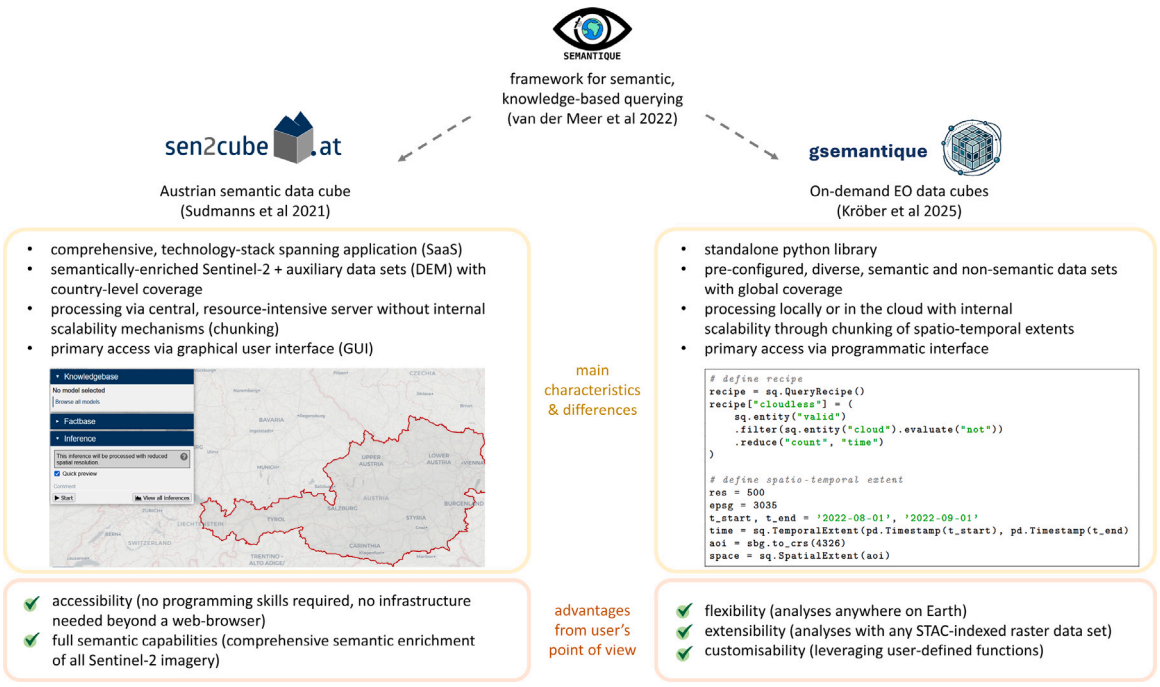

Fig. 1. Relationship of our work to other semantic, knowledge-based EO analysis systems.

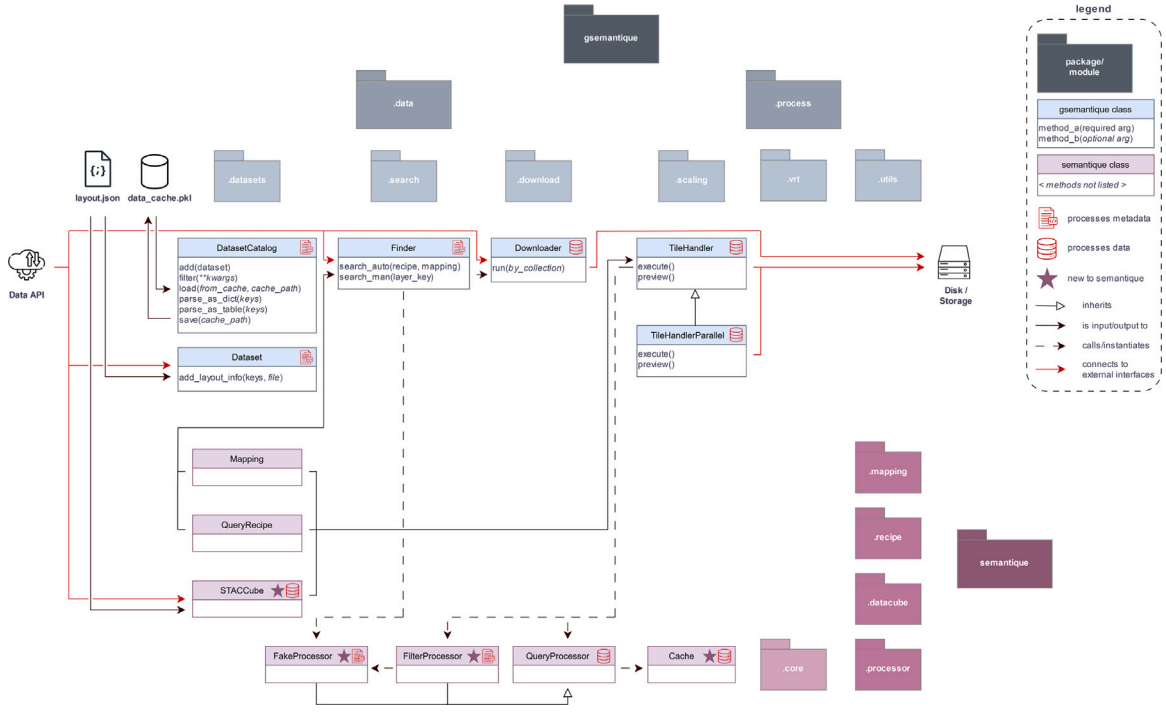

Fig. 2. Implementation of our on-demand EO cube architecture with package structure and main classes.

(requirements (A)) and efficient processing at scale (requirements (C)). Both packages are lightweight Python libraries fulfilling the general software properties as formulated in requirements (D). From a technical point of view, the core structure of both packages and their relationship is depicted in Fig. 2. Note that for *gsemantique* all classes are shown, whereas for *semantique* only a selection of relevant classes is shown to keep the figure clear and concise. The precise translation of the first three groups of requirements (A)-(C) into architectural design choices as shown in Fig. 2 is described subsequently.

In terms of data access (A.1–A.3), STAC is used as a widely accepted metadata standard to manage data description and retrieval in a consistent manner. The *STACCube* class implemented in *semantique* allows to create data cubes based on item collections as results of STAC metadata searches. The metadata searches themselves are encapsulated in the *Finder*. To search for STAC items, catalog endpoints and collection names need to be given. These are organized together with other metadata as *data set* objects added to a *DatasetCatalog*. A predefined *DatasetCatalog* is stored within *gsemantique* (A.1). Currently, it covers references to a total of 13 data sets (collections) with

Table 1
Data sets with global coverage ready-to-use in *gsemantique* via predefined data set objects and layout file. This list can be extended with any STAC-indexed data set.

Data set information			Data layer information			Temporal information	
STAC collection	**STAC catalog**	**Category**	**# Layers**	**Non-Semantic ones**	**Semantic ones**	**Extent**	**Frequency**
sentinel-1-rtc	[A]	SAR	4	Amplitude in four polarizations	–	2014 (Oct) – today	Sub-daily
sentinel-1-global-coherence	[B]	SAR	17	6-, 12-, 18-, 24-, 36-, 48-day coherence in VV and VH polarizations	–	2020	Quarterly
sentinel-2-l2a	[A]	Multispectral	13	Twelve reflectance bands	Scene Classification Map (SCM)	2015 (June) – today	Sub-daily
sentinel-2-l2a	[C]	Multispectral	13	Twelve reflectance bands	SCM	2015 (June) – today	Sub-daily
landsat-c2-l2	[A]	Multispectral	10	Nine reflectance bands	Quality assessment band	1982 (Aug) – today	Sub-daily
esa-worldcover	[A]	Landcover	1	–	10-class LULC layer	2020–2021	Yearly
io-lulc-annual-v02	[A]	Landcover	1	–	9-class LULC layer	2017–2023	Yearly
nasadem	[A]	DEM	1	Elevation layer	–	2000	Static
cop-dem-glo-30	[A]	DSM	1	Elevation layer	–	2010–2015	Static
modis-64A1–061	[A]	Fire Detection	3	Ordinal burn date, burn date uncertainty	Quality assessment band	2000 (Nov) – today	Monthly
modis-14A2–061	[A]	Fire Detection	2	–	Categorization of fire confidence, quality assessment band	2000 (Feb) – today	Monthly
jrc-gsw	[A]	Hydrogeography	4	Water frequency, frequency changes	Binary water existence, categorical changes in surface water status	1984–2020	Annual
glo-30-hand	[B]	Hydrogeography	1	Height above nearest drainage layer	–	2010–2015	Static

[A] https://planetarycomputer.microsoft.com/api/stac/v1, [B] https://stac.asf.alaska.edu, [C] https://earth-search.aws.element84.com/v1

more than 70 individual bands (assets) as shown in Table 1. Using predefined methods to add new data sets, the *DatasetCatalog* can be extended flexibly to include any data set for which a STAC catalog endpoint and collection name can be specified (A.2). This is not limited to dynamic STAC catalogs but also includes static ones. Users can therefore address a wide range of additional, publicly available data sets (https://stacindex.org) as well as local ones as long as STAC-conformant metadata is available. Importantly, *gsemantique* has another object containing data set information, which is the layout json file that is used to initialize the *STACCube*. This layout file contains metadata not on the data set level but the individual data layers (i.e. the assets). It is relevant to guide data fetching since it defines data types, value ranges and missing data values. Together, the predefined *DatasetCatalog* and layout file structure the variable parts of EO data and define which data is accessible for subsequent data cube construction. Finally, storage and persistence functionality for the input data is realized via the *Downloader* (A.3). It leverages a library for asynchronous, non-blocking I/O tasks to efficiently manage the high-concurrency operation of fetching data for the metadata search results as obtained by the *Finder*, and transfer it to a specified location. The data is automatically STAC-indexed by building a static STAC catalog and collection for seamless integration in further analyses including data cube construction.

Regarding analysis support (B.1–B.5), *semantique* (Van Der Meer et al., 2022) already provided a strong basis for standard data cube operations (e.g. aggregation via reduce-through-space/time) (B.1), modeling semantic concepts (B.2), and representing expert knowledge (B.3). These core components of the modeling are implemented using corresponding predefined structures (mapping and recipe) as described in Van Der Meer et al. (2022) and again briefly outlined in Section 4. The set of predefined modeling functions, which can be used within these structures, can be flexibly extended by user-defined functions. This is ensured by corresponding interface functions that enable the integration of any standard Python code (B.4). Extensions of this existing functionality mainly concern two points: The first one targets effective communication and exchange about analysis workflows by adding visualization options to represent semantic models graphically (B.5). Following Sudmanns et al. (2021), we rely on the JavaScript library Blockly (Google, 2024) with custom definitions of visual blocks for corresponding representations. The second extension of *semantique*, covers the increased complexity of translating semantic models into procedural code for on-demand cubes. The difficulty here is that the user defines a semantic model (e.g. using entities such as clouds), while the implementation requires to resolve the underlying data layer references (e.g. SCM of S-2) along with their queried spatiotemporal extent. Indeed, semantic querying on static, persistent data cubes such as ODC instances also requires this translation into numerical code. However, unlike their more comprehensive counterparts, on-demand cubes lack the underlying native capabilities for efficient retrieval of data, e.g. via data base indexing. This problem is exacerbated by the fact that the user can query data from the comprehensive totality of all EO data worldwide and the data are not necessarily stored on high-performance infrastructures with low retrieval latencies. A feasible implementation of semantic, on-demand data cubes therefore requires the metadata for constructing the data cubes to be narrowed down as far as possible in advance in order to speed up the subsequent actual data retrieval. This task is carried out by *FakeProcessor* and *FilterProcessor* objects. *FakeProcessor* instances resolve all semantic concepts in a model into data layer references prior the actual model evaluation. *FilterProcessor* instances evaluate the required temporal extents for which the referenced data must be loaded according to the model. If a model contains temporal filter operations for individual data layers, these are resolved using *FilterProcessor* before the actual data fetching. *FakeProcessor* and *FilterProcessor* together enable the user to perform modeling completely at the semantic level while ensuring an efficient translation into numerical code.

Finally, processing requirements concerning scalability and efficiency with abstracted complexity (C.1–C.3) are implemented in *gsemantique* via the *TileHandler* and *TileHandlerParallel*. Enabling scalability in a hardware-independent way, i.e. not relying on vertical scaling,

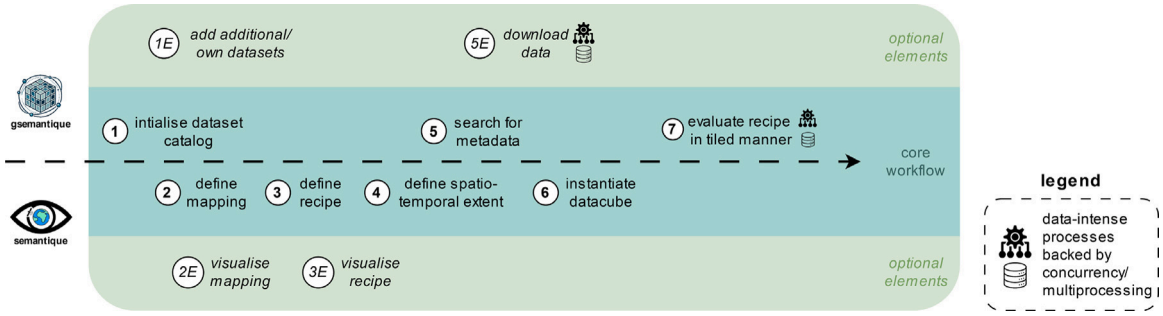

Fig. 3. Generalized end-to-end EO analysis workflow using *gsemantique*.

is realized by tiling the overall data cube into smaller chunks to run the code sequentially on them with a final merger of the chunked results (C.1). This is implemented by the *TileHandler* class. It analyses the recipe for operations executed on the spatial and/or temporal dimension and chooses the remaining one as a chunking dimension. A preview is calculated containing information on the number of chunks to be processed and the expected size of the output. Instead of sequential processing of the chunks, parallelization via multicore processing is possible using the *TileHandlerParallel* class (C.2). In both cases, the data for each chunk is cached using an instance of the *Cache*, ensuring that multiple requests of the same data within a semantic model are handled with an efficient one-time data fetching process (C.2). The chunked results are finally merged as single outputs or optionally virtual rasters in case of spatial outputs. Opting for virtual rasters enables storage-efficient processing of non-rectangular areas of interest. The overall processing complexity is abstracted from the user by full automation of the split-apply-merge workflow based on reasonable default values, e.g. for spatial and temporal chunk sizes (C.3).

4. System utilization

Using *gsemantique* to conduct EO analysis with on-demand data cubes is a matter of a few lines of Python code. The general workflow fundamental to any analysis is outlined in Fig. 3 and an example on how to translate this into code for a specific analysis is demonstrated in Fig. 4.

The seven stage workflow starts with the initialization of the *DatasetCatalog*. Usually, this amounts to reading the metadata for all predefined data sets as listed in Table 1. However, if a user wants to add custom data sets to be employed in the following analysis, the *DatasetCatalog* can be instantiated based on any custom layout file that extends the predefined data sets. The subsequent definition of the analysis model, is split into two parts. First, the definition of a *Mapping* as a collection of entities is carried out. The entities represent semantic concepts defined by a set of properties. In the case shown in Fig. 4, the entities are ‘valid observations‘ and ‘clouds‘, both being defined by their corresponding classification according to the S-2 Level 2 A SCM. More advanced cases, where the properties that define entities are not immediately drawn from already existing classifications, are shown in Section 5.2. Second, the definition of a *Recipe* as the application part is required. The entities of interest are transformed via spatiotemporal processing and other data cube operations to derive analysis results. Both parts, *Mapping* and *Recipe* represent Python dictionaries, but can be visualized in a graphical block structure (Fig. 7). The fourth stage of the workflow requires the user to define the area and time frame of interest to select the spatiotemporal extent for which the analysis should be carried out. Given this selection and the data references as encoded in the *Mapping* and *Recipe*, the user can query the metadata of all EO files necessary to calculate the results. Optionally, the user can query the raster data and download them, to persist the data on the disc. The sixth stage is the instantiation of the *STACCube*, which takes the previously acquired metadata with pointers to the data locations as an input. The creation of the data cube object itself does not load any data. The same lazy-loading strategy applies to the instantiation of the *TileHandler* in the final stage, where the general structure to calculate results is set up based on the previously defined *Mapping*, *Recipe*, *STACCube*, spatiotemporal extent. With a single call to execute the processing, the *TileHandler* takes care of calculating the results while abstracting the technical complexity of chunking the data cube with sizes that fit into the main memory during processing. If desired, the user can decide to tune the default chunk sizes along the spatial or temporal dimension by specifying them as arguments upon the instantiation of the *TileHandler* object.

5. Demonstration

The following application cases demonstrate the strength and flexibility of the on-demand data cube approach for semantic, knowledge-based querying of EO data. A summary of the use cases and their specific objectives is provided in Table 2.

5.1. Application I – cloud-free scenes

Frequently, clouds are obscuring the Earth’s surface, complicating analyses based on optical remote sensing. Large parts of Europe, western North America, and areas within the equatorial low-pressure trough are characterized by cloud frequencies greater than 50% (Wilson and Jetz, 2016). Spatial information about the frequencies of cloud-free observations for a given satellite can be leveraged to anticipate possible complications in applying models in areas with few available cloud-free scenes, or to select areas rich of cloud-free observations as promising study areas. A corresponding evaluation on the availability of cloud-free scenes for S-2 is presented by Sudmanns et al. (2020a), for example. However, this analysis is based on scene-wide metadata on cloud coverage. With semantically enriched data, where label information on the existence of clouds is available at the pixel level, such analyses can be carried out with increased spatial precision. Beyond semantically enriched data availability, the prerequisites encompass a modeling framework that supports the construction of semantic queries. Furthermore, a processing engine allowing efficient and scalable computations is required since aggregating cloud-free observations over time on the pixel level is a resource- and data-intensive process (Table 2). Finally, the task is well-suited to be solved via on-demand data cubes as users may want to retrieve cloud-coverage data once to select their study area of interest without the need for extensive area-wide computations justifying the effort to setup a persistent, more comprehensive data cube framework. As shown in Fig. 4, conducting such analyses is a relatively simple task using *gsemantique*. Applying the script in a slightly modified version on a continental scale leads to the results as depicted in Fig. 5.

Conclusions that can be drawn based on Fig. 5 about the availability of cloud-free observations include spatially precise details, e.g. on the recording geometry of S-2 data in strips as well as topographical effects such as orographic cloud formation. It should be noted that, on a

```python
# general imports
import geopandas as gpd
import json
import os
import pandas as pd
import semantique as sq
import gsemantique as gsq

# step 1: load data catalog
ds_catalog = gsq.DatasetCatalog()
ds_catalog.load()

# step 2: define mapping
# i.e. relationship semantic concepts <-> numeric values
mapping = sq.mapping.Semantique()
mapping["entity"] = {}
mapping["entity"]["valid"] = {
    "class": (
        sq.layer("Planetary", "classification", "scl")
        .evaluate("not_equal", 0)
    )
}
mapping["entity"]["cloud"] = {
    "class": (
        sq.layer("Planetary", "classification", "scl")
        .evaluate("in", [8, 9, 10])
    )
}

# step 3: define recipe
# i.e. processing of semantic concepts
recipe = sq.QueryRecipe()
recipe["cloudless_count"] = (
    sq.entity("valid")
    .filter(sq.entity("cloud").evaluate("not"))
    .reduce("count", "time")
)
```

```python
# step 4: define spatio-temporal extent
epsg = 3035 # coordinate reference system (CRS)
res = 500 # resolution in CRS units
t_start, t_end = '2022-01-01', '2023-01-01'
time = sq.TemporalExtent(
    pd.Timestamp(t_start),
    pd.Timestamp(t_end)
)
aoi = gpd.read_file("polygon.geojson").to_crs(4326)
space = sq.SpatialExtent(aoi)

# step 5: search for metadata
fdr = gsq.Finder(
    ds_catalog,
    t_start,
    t_end,
    aoi
)
fdr.search_auto(recipe, mapping)

# step 6: instantiate datacube
with open(gsq.LAYOUT_PATH, "r") as file:
    dc = sq.datacube.STACCube(
        json.load(file),
        src = fdr.item_coll
    )

# step 7: evaluate recipe
# create TileHandler instance & execute processing
context = dict(
    recipe = recipe,
    datacube = dc,
    mapping = mapping,
    space = space,
    time = time,
    spatial_resolution = [-res, res],
    crs = epsg
)
th = gsq.TileHandlerParallel(n_procs = 12, **context)
th.execute()
```

Fig. 4. Code example for end-to-end EO analysis workflow using *gsemantique*.

Table 2
Summary of use cases realized via *gsemantique*.

Use case description		Analysis extent		Processed input data[a]		Design rationale & aim
Main focus	**Subpart**	**Space**	**Time**	**Number of scenes**	**Download volume**	
Cloud-free scenes	Evaluate number of cloud-free observations through time	Europe (excl. French overseas territories), 5.837.000 km^2	2021–2023	S-2: 872,541	S-2: 1128.75 GB	• Highlight the benefits of basic semantic querying capabilities in an EO data cube framework • Showcase scalability possibilities and limits
	Create monthly cloud-free composites	Lower Austria, 19.200 km^2	2022	S-2: 1475	S-2: 773.89 GB	
Forest disturbances	Analyze magnitude and persistence of forest disturbances	Lake Irrsee at the border between Salzburg and Lower Austria, 78.5 km^2	2020–2023	ESA Worldcover: 1 DEM: 1 S-1: 8 S-2: 564	ESA Worldcover: 84.8 MB DEM: 18.7 MB S-1: 14.7 MB S-2: 945 MB	• Demonstrate possibilities for semantic modeling beyond using predefined entities by defining own entities using multimodal queries • Showcase flexibility to define and compare different models based on different data sets & modeling assumptions

[a] *The number of scenes and the download volume both depend on the native format of the data. The number of scenes is calculated as the sum of the unique files (e.g. for S-2, the sum of the unique product URIs), while the download volume includes the size of all bands that are actually processed (e.g. for S-2, the SCM for the cloud statistics, and the R,G,B & NIR bands for the cloud composites).*

consumer-grade infrastructure, the calculations for Fig. 5 can only be created on a monthly basis and aggregated post-hoc annually due to the northern areas with multiple overlapping orbits causing a high dimensionality of the chunks with more than a thousand observations in the temporal domain. For local calculation on consumer-grade infrastructure, the analysis for a single month takes several hours, as large amounts of data (Table 2) have to be transferred and processed. Efficiency gains can be realized by deploying *gsemantique* on well-equipped cloud infrastructures close to the data servers. A detailed comparison of total model execution times under varying cloud server configurations can be found in the supplementary material of this article.

Beyond the calculation of cloud coverage statistics, the usability of cloud cover information also concerns filtering data with high cloud coverage as a pre-processing step in many EO processing workflows. A common technique is the creation of cloud-free composites as a higher level, analysis ready data product for subsequent processing. Creating such composites via semantic querying involves the pixel-based exclusion of all cloudy observations, conducting a temporal aggregation of reflectances for all observations that are flagged as non-cloudy. Using the SCM accompanying S-2 Level 2 A data for the definition of cloud entities, exemplary results for this simple semantic approach are shown in Fig. 6 (proposed approach). This can be contrasted with the non-semantic median composite as an alternative approach. This compositing technique is based on statistical assumptions on how clouds can be filtered indirectly, namely that the median reflectance of an entire time series is likely to represent cloud-free conditions (Fig. 6, baseline a). If scene-wide metadata on cloud-coverage is available, it can be used to further enhance the applicability and robustness of the median compositing by pre-filtering scenes with low cloud coverage, building the median only among those pre-filtered scenes (Fig. 6, baseline b).

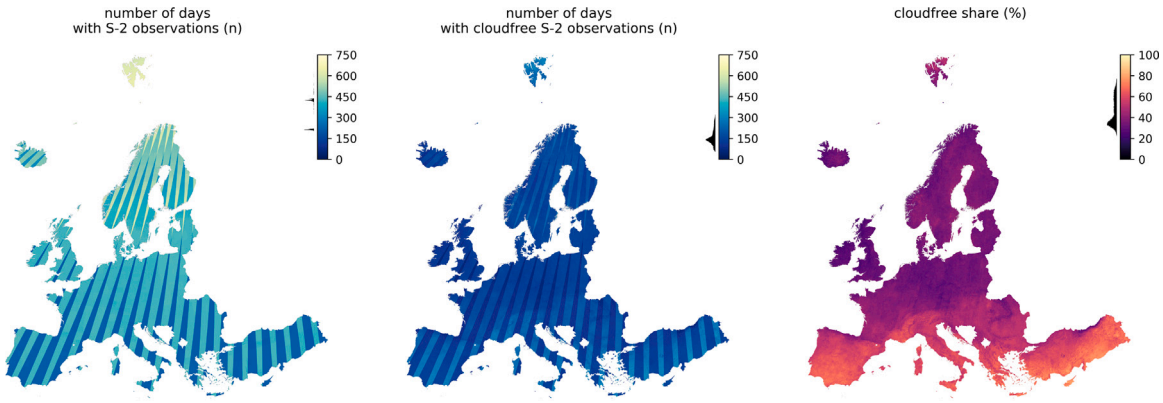

Fig. 5. Availability of S-2 observations over Europe for the years 2021–2023. The proportion of cloud-free observations shown on the right represents the ratio between the middle and left subfigure.

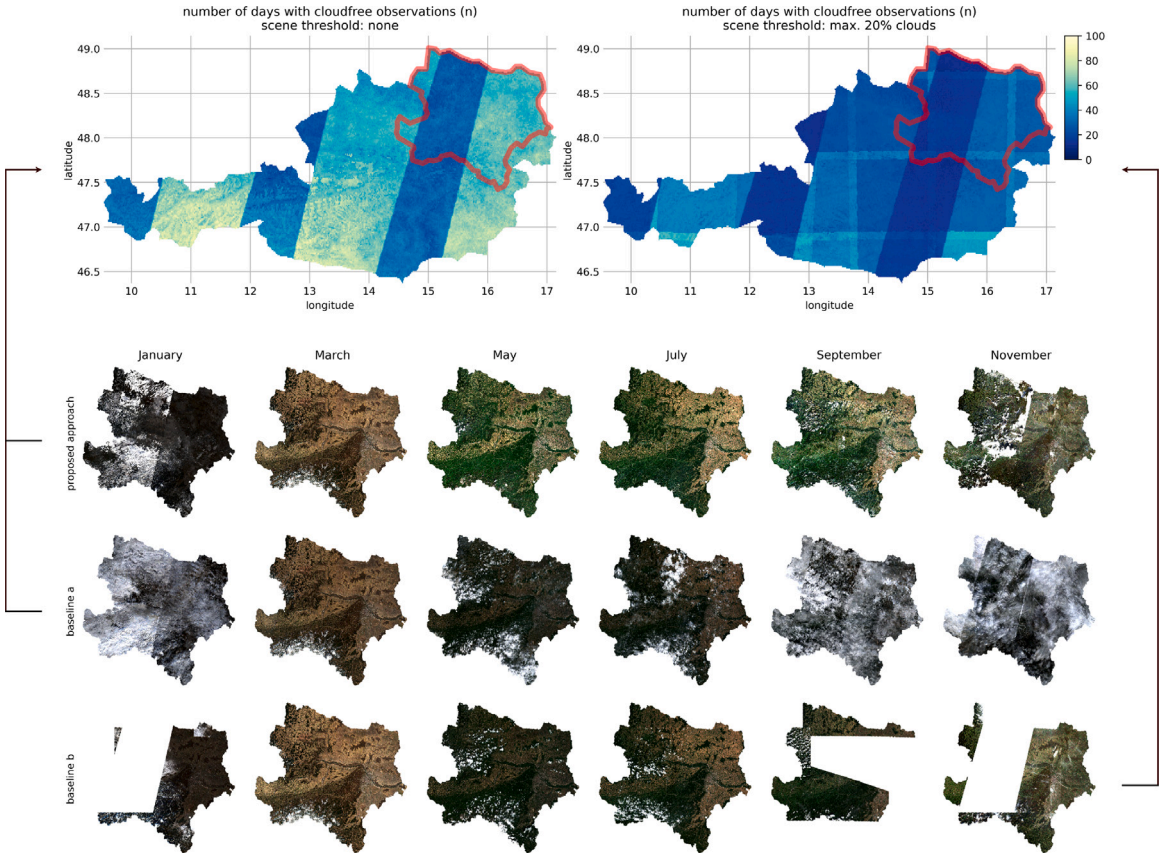

Fig. 6. Cloud-free composites using semantic and non-semantic median compositing. The upper subfigures indicate the number of observations available for composite formation. For a detailed description of the compositing techniques themselves (lower subfigures), refer to the explanations in Section 5.1.

As evident from the true color RGB visualizations for a selection of six out of twelve monthly composites, the semantic approach leads to superior results in terms of a consistent exclusion of clouds on a pixel-level. Under very cloudy conditions, the non-semantic median composite is natively incapable of selecting cloud-free observations. Pre-filtering the scenes reduces the number of available observations, possibly leading to a complete lack of observations for a given month. The observation is neither surprising nor is the applied semantic compositing technique novel. The point here is that the example clearly illustrates the effectiveness and relevance of a knowledge-based querying framework capable of exploiting the full amount of semantically enriched observations in a spatiotemporal data cube.

5.2. Application II – forest disturbances

Assessing the status of forests via remote sensing is a common research and application field with partially overlapping branches of investigating forest degradation (Hirschmugl et al., 2017; Gao et al., 2020), forest disturbance (Frolking et al., 2009; Hirschmugl et al., 2017; Paulino et al., 2024) and forest health (Lausch et al., 2016, 2017). With our use case, we follow Paulino et al. (2024) in a defining forest disturbances as natural or anthropogenic events resulting in forest changes that are identifiable by means of EO.

The task of forest disturbance mapping has three distinct aspects that makes it well-suited to be tackled with our framework: Firstly,

forest disturbances are inherently process-based, i.e. they are characterized by a temporal change in the forest's status. A data cube-based approach is therefore well positioned, as it allows to query every single observation through time. Secondly, the task is characterized by terminological complications with the existence of slightly different definitions of the phenomenon to be modeled. Remote sensing experts thus need to make their specific modeling assumptions explicit and transparent to enable meaningful exchange with others. The definition issues together with difficulties in acquiring ground truth data also cause a lacking availability of homogeneous label data sets impeding data-driven ways to solve the task. A semantic, knowledge-based modeling approach that allows to create a set of human-readable models based on different modeling assumptions is therefore a viable alternative. The third and final aspect of forest disturbance modeling is that neither the entity of interest, i.e. the forest, nor the process, i.e. the forest's change, are straightforward to be modeled. The entity and process of interest can be characterized by more than one property, which may require multimodal data usage to combine sensory information. Our framework provides explicit means to connect physical world and numerical views by mapping multiple properties of an entity in the semantic domain to features of the object in the image domain, leveraging a diverse set of predefined data sets for feature definition. For demonstration purposes, we generate a model suite with a total of four models based on two entity definitions of forest in an undisturbed state crossed with two process definitions that model potential disturbances (Fig. 7).

The entity definitions showcase two possibilities to calculate the extent of forests either in a data-driven or knowledge-driven manner. For the former, we simply equate forests with the tree cover class of ESA Worldcover (Zanaga et al., 2021), which has been generated via a gradient boosting decision tree applied to multimodal data. For the expert-knowledge-based definition, we create a set of forest properties with a corresponding mapping to numerical representations by relying on our own knowledge but also insights of former remote sensing studies on forests. We assume forests to be characterized by temporal stability translated to low radar coherence (Jacob et al., 2020; Nikaein et al., 2021; Borlaf-Mena et al., 2021), complex structure translated to a low-to-medium radar backscatter intensity (Dostálová et al., 2018; Nikaein et al., 2021; Borlaf-Mena et al., 2021) and an altitude below the tree line translated to an elevation below a thresholded elevation level (Hagedorn et al., 2006).

The two process definitions showcase how different modeling assumptions regarding the phenomenon of interest can be integrated. Both process definitions are based on the annual proportion of vegetation counts as given by the S-2 SCM, which is taken as a simplified proxy for forest vitality in a given year. Starting with the vegetation proportion statistics representative of the undisturbed forest state in the first year (Fig. 7, *status_original*), a comparison with the vegetation proportion statistics for the following years (Fig. 7, *status_post*) is carried out. Vegetation proportion decreases beyond a certain threshold are counted as relevant changes, which are then used to define disturbance magnitude and persistence as two exemplary custom properties of interest. The two process definitions differ in their threshold values as well as reducer functions used to calculate magnitude and persistence. The reducer function for persistence, for example, is either counting every year in which the threshold has been exceeded (Fig. 7, *sensitive model*), or only the amount of consecutive years with vegetation decreases larger than the threshold (Fig. 7, *robust model*).

The results of applying the models to a forested area close to Irrsee, Austria, analyzing forest disturbances in a four-year period are shown in Fig. 8. For the forest extent delineation, it is evident that both entity definitions correctly identify the central large forest areas as such. However, the ESA Worldcover definition additionally includes many smaller areas as fine-grained forest patches, which roots back to the fact that the class definition used actually identifies trees on a pixel-basis rather than larger spatially contiguous forest patches. The knowledge-based definition has a stronger tendency to spatial generalization given its foundation of Sentinel-1 coherence and backscatter data with coarser spatial resolution. The comparison of the model results thus reflects the different modeling assumptions made, which could now be used in an iterative process to refine the models. Given the semantic and visualizable design of the models, domain experts can be easily integrated in this discussion and adjustments of models. The ease to create and compare multiple models based on different data sources and methods offers further potential to estimate the degree of uncertainties in the final results, while leveraging the unified suite of models as an ensemble model with increased robustness. This not only applies to the entity definitions but equally to the process definitions with the results of forest disturbance magnitude and persistence.

It is worth emphasizing that the focus of this application example is not to create an optimal model achieving state-of-the art results in competition with other, more elaborated study designs. Intentionally, a conceptually rather simple model design was chosen to focus on highlighting the conceptual advantages brought by our data cube approach in facilitating semantic, knowledge-based analysis in line with the goals defined in Table 2. In terms of accuracy, the only aim is to showcase that our models lead to reasonably realistic results. This is given for both, forest extent delineation and disturbance assessments, as can be confirmed visually by comparing the modeling results in Fig. 8 to the true-color RGB timeseries visualizations. Moreover, the results are largely in line with data obtained from the European Forest Disturbance Atlas (Viana-Soto and Senf, 2025), providing further evidence for the basic reasonability of our model. The model can be easily refined and adapted to a specific target application for flexible use anywhere in the world.

6. Limitations & outlook

In terms of software implementation, further improvements concerning aspects of scalability are envisioned. Currently, features of parallelization and scheduling are performed on a chunk level with uniform chunk dimensionality as described in Section 3.2. This rigid tiling scheme with parallelization achieved on a high-level works but can be improved to achieve higher efficiency. Data of neighboring chunks are likely to be loaded multiple times as the tiling schema is not optimized with regard to the spatiotemporal extents of the native files. Caching, currently realized only within chunks but not across chunks could offer potential to mitigate redundancy in querying data. Still, sequentialization or parallelization of processing on a chunk level, where the execution of full models for a given chunk is considered the smallest unit, remains sub-optimal for at least two reasons. Firstly, this approach is inherently limited when a model involves both spatial and temporal operations that make model-wide chunking along the space or time axis impossible. Secondly, parallelization at the lower level of the individual functions within a model enables a better, more even utilization of the processing resources. To this end, we consider the integration of the Dask framework (Dask Development Team, 2016) as a promising way to improve the current implementation by relying on an established standard for the efficient parallelization of array-based processing. Prioritizing user-friendliness, we currently do not use Dask in our framework. The flexible creation of complex models requires a thorough understanding of task graph optimization on the user's side in order to exploit potential efficiency benefits of the full task scheduling provided by Dask.

A second point on extending software functionality refers to stronger support for multimodal data fusion. As of today, *gsemantique* offers predefined data set connections and means to integrate different data sets during the modeling process. The currently prevailing way of integrating multimodal data sets is to map different sensory information to properties of entities as demonstrated in Fig. 7. This data integration approach is primarily relevant for data sets derived from a heterogeneous set of sensors acquiring different information

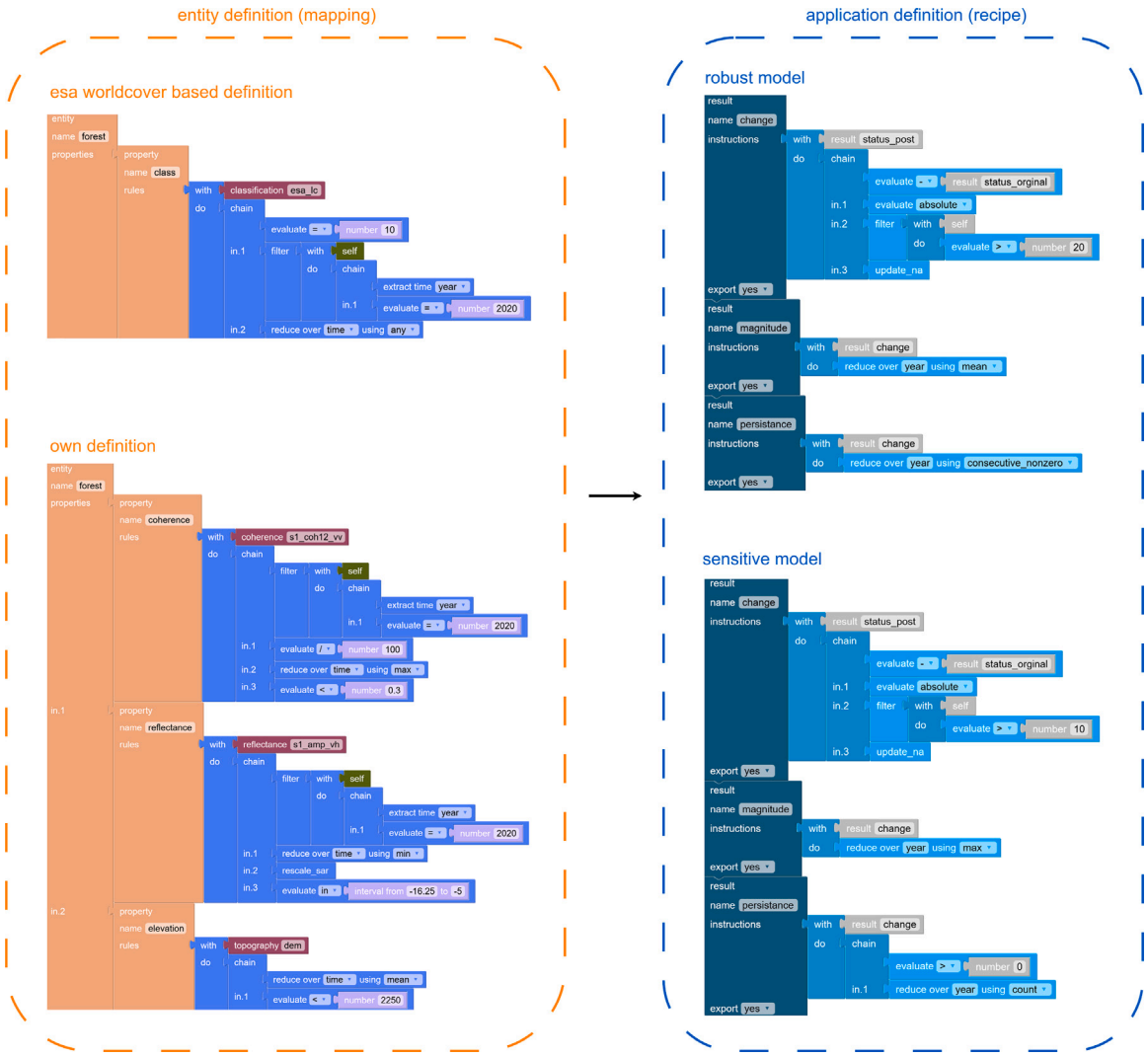

Fig. 7. Semantic models used for the assessment of forest disturbance.

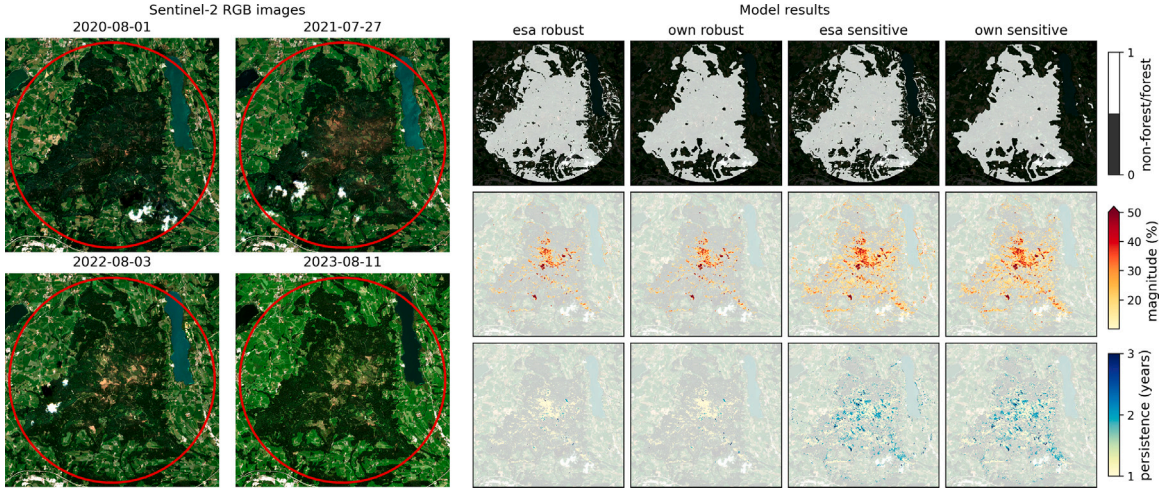

Fig. 8. Timeseries of true-color RGB visualizations of the area of interest (left) and forest disturbance maps (right). For the different underlying entity and process definitions, refer to Fig. 7.

about the entities of interest. For more homogeneous data sets, such as S-2 and Landsat with a large degree of overlapping bands, additional support for unifying data to model the same entity property would be beneficial. Corresponding automated routines for data harmonization are, for example, provided by Frantz (2019).

As a final point regarding software improvements, we would like to foster the better integration of data-driven and knowledge-based workflows. Demonstrated in Section 5.2, *gsemantique* allows to leverage data-driven modeling by integrating data layers as input, which in turn have been produced using machine learning. Furthermore, one can leverage user-defined functions to integrate machine learning or other data-driven elements during the model definition. Therefore, while our framework provides strong support for knowledge-based analysis, it does not exclude data-driven modeling. Still, the explicit integration of both domains could be improved in a way that machine and deep learning practitioners could seamlessly integrate data cubes prepared in a knowledge-based manner into their workflows and vice versa. Corresponding developments could be seen as complementary to the directions already indicated elsewhere with regard to improving data-driven modeling in EO cubes (Montero et al., 2024b).

Looking ahead on the more application-related future works, there are several interesting directions in the context of which our new framework could be explored. One of these concerns the question of analyzing synergies between multimodal data usage and semantic, knowledge-based modeling. With mono-modal data, only a narrow set of the properties of physical world entities can be modeled, which is different if multimodal data is available. Suggested by Bahmanyar et al. (2015), increasing the diversity of data sets in analyses could help to close the semantic gap as the difference between the users and the computers view on a given entity. Using *gsemantique*, one can model multiple different perspectives on the same entity, drawing on a variety of data sets and their possible combinations, in order to explore corresponding hypotheses in greater detail. Linked to this, ideas for further research designs include the analysis of uncertainty in modeling. In Section 5.2, we have indicated the potential for such analyses using multiple definitions of the forest entity and the phenomenon of forest disturbance. The possibility of integrating one's own data into the analysis in *gsemantique* allows for far-reaching comparisons to be made as to how far modeling results depend on the data basis and the combination of data in a model.

7. Conclusion

This work was motivated by the identified need for new frameworks that mitigate the discrepancy between extensive data availability and restricted possibilities of analyzing them to achieve big EO image understanding. Reviewing the current state of the art, we noted that the variety of existing frameworks do not cover the requirements for ad-hoc, mesoscale analyses, which are commonly conducted in practice. Whereas such analyses exceed standard resources in terms of local hardware limitations for data processing, and require the shift to modern data cube- and cloud-based processing paradigms, they are not justifying the effort to set up complex persistent infrastructures. Additionally, approaches supporting the actual analysis of data, i.e. the modeling process that integrates different data sets to move from the non-semantic data level to condensed information, are integrated in existing data cube approaches insufficiently. The prevailing focus so far has been on technical means of data access, possibly extended by data-driven means of modeling. As a consequence of these deficiencies, we extended an existing semantic querying language towards an on-demand EO data cube system with end-to-end support for the whole EO analysis workflow. We outlined the implementation of the proposed system focusing on its main functionalities, which are pre-configured, extensible access to a variety of EO data sets with global coverage, strong analysis support through a framework for semantic, knowledge-based image understanding, and processing support to scale analyses in space and time. Those properties have been demonstrated by two application examples, both designed in a conceptually simple yet effective way to derive useful information from EO data in a few lines of code. Our work thus makes a valuable contribution to foster the structured transformation of data into condensed information with the aims of enabling analytical insights, supporting decision-making and generating further EO knowledge.

CRediT authorship contribution statement

Felix Kröber: Writing – original draft, Visualization, Software, Formal analysis, Conceptualization. **Martin Sudmanns:** Writing – review & editing, Project administration, Funding acquisition, Conceptualization. **Lorena Abad:** Writing – review & editing, Software. **Dirk Tiede:** Writing – review & editing, Validation, Supervision, Project administration, Funding acquisition.

Funding

The research leading to these results has received funding from the European Union's Horizon Europe research and innovation program under the Grant Agreement No. 101082493 (Project: LEONSEGS).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank Luuk van der Meer for his support in integrating the adaptations of the semantique framework. Furthermore, the authors gratefully acknowledge the reviewers for their comments which have contributed to the improvement of this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.isprsjprs.2025.07.015.

Data/code availability statement

- name of the software: gsemantique
- hard-/software requirements: any OS (Windows, Mac or Linux); Python installation
- source code availability: https://github.com/Sen2Cube-at/gsemantique
- analysis data availability: https://doi.org/10.5281/zenodo.15423258

References

Ackoff, R.L., 1989. From data to wisdom. J. Appl. Syst. Anal. 16 (1), 3–9.

Appel, M., Pebesma, E., 2019. On-Demand Processing of Data Cubes from Satellite Image Collections with the gdalcubes Library. Data 4 (3), 92. http://dx.doi.org/10.3390/data4030092.

Ariza-Porras, C., Bravo, G., Villamizar, M., Moreno, A., Castro, H., Galindo, G., Cabera, E., Valbuena, S., Lozano, P., 2017. CDCol: A geoscience data cube that meets colombian needs. In: Advances in Computing. CCC 2017. Communications in Computer and Information Science. 735, Springer, pp. 87–99. http://dx.doi.org/10.1007/978-3-319-66562-7_7.

Arvor, D., Belgiu, M., Falomir, Z., Mougenot, I., Durieux, L., 2019. Ontologies to interpret remote sensing images: why do we need them? GIScience Remote. Sens. 56 (6), 911–939. http://dx.doi.org/10.1080/15481603.2019.1587890.

Arvor, D., Betbeder, J., Daher, F.R., Blossier, T., Le Roux, R., Corgne, S., Corpetti, T., De Freitas Silgueiro, V., Silva Junior, C.A.D., 2021. Towards user-adaptive remote sensing: Knowledge-driven automatic classification of Sentinel-2 time series. Remote Sens. Environ. 264, 112615. http://dx.doi.org/10.1016/j.rse.2021.112615.

Asmaryan, S., Muradyan, V., Tepanosyan, G., Hovsepyan, A., Saghatelyan, A., Astsatryan, H., Grigoryan, H., Abrahamyan, R., Guigoz, Y., Giuliani, G., 2019. Paving the Way towards an Armenian Data Cube. Data 4 (3), 117. http://dx.doi.org/10.3390/data4030117.

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

Bahmanyar, R., Murillo Montes De Oca, A., Datcu, M., 2015. The Semantic Gap: An Exploration of User and Computer Perspectives in Earth Observation Images. IEEE Geosci. Remote. Sens. Lett. 12 (10), 2046–2050. http://dx.doi.org/10.1109/LGRS.2015.2444666.

Baltsavias, E., 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. ISPRS J. Photogramm. Remote Sens. 58 (3–4), 129–151. http://dx.doi.org/10.1016/j.isprsjprs.2003.09.002.

Baraldi, A., 2011. Satellite Image Automatic Mapper - a Turnkey Software Executable for Automatic Near Real-Time Multi-Sensor Multi-Resolution Spectral Rule-Based Preliminary Classification of Spaceborne Multi-Spectral Images. Recent. Patents Space Technol. 1 (2), 81–106. http://dx.doi.org/10.2174/1877611611101020081.

Baraldi, A., Boschetti, L., 2012. Operational Automatic Remote Sensing Image Understanding Systems: Beyond Geographic Object-Based and Object-Oriented Image Analysis (GEOBIA/GEOOIA). Part 1: Introduction. Remote. Sens. 4 (9), 2694–2735. http://dx.doi.org/10.3390/rs4092694.

Baraldi, A., Sapia, L.D., Tiede, D., Sudmanns, M., Augustin, H.L., Lang, S., 2023. Innovative Analysis Ready Data (ARD) product and process requirements, software system design, algorithms and implementation at the midstream as necessary-but-not-sufficient precondition of the downstream in a new notion of Space Economy 4.0 - Part 1: Problem background in Artificial General Intelligence (AGI). Big Earth Data 7 (3), 455–693. http://dx.doi.org/10.1080/20964471.2021.2017549.

Baumann, P., 2017. The Datacube Manifesto. URL: https://earthserver.eu/tech/datacube-manifesto/The-Datacube-Manifesto.pdf.

Baumann, P., Mazzetti, P., Ungar, J., Barbera, R., Barboni, D., Beccati, A., Bigagli, L., Boldrini, E., Bruno, R., Calanducci, A., Campalani, P., Clements, O., Dumitru, A., Grant, M., Herzig, P., Kakaletris, G., Laxton, J., Koltsida, P., Lipskoch, K., Mahdiraji, A.R., Mantovani, S., Merticariu, V., Messina, A., Misev, D., Natali, S., Nativi, S., Oosthoek, J., Pappalardo, M., Passmore, J., Rossi, A.P., Rundo, F., Sen, M., Sorbera, V., Sullivan, D., Torrisi, M., Trovato, L., Veratelli, M.G., Wagner, S., 2016. Big Data Analytics for Earth Sciences: the EarthServer approach. Int. J. Digit. Earth 9 (1), 3–29. http://dx.doi.org/10.1080/17538947.2014.1003106.

Baumann, P., Misev, D., Merticariu, V., Huu, B.P., Bell, B., 2018. Rasdaman: Spatio-temporal datacubes on steroids. In: Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, Seattle Washington, pp. 604–607. http://dx.doi.org/10.1145/3274895.3274988.

Belgiu, M., Drăguţ, L., 2016. Random forest in remote sensing: A review of applications and future directions. ISPRS J. Photogramm. Remote Sens. 114, 24–31. http://dx.doi.org/10.1016/j.isprsjprs.2016.01.011.

Berners-Lee, T., Hendler, J., Lassila, O., 2001. The Semantic Web. Sci. Am. 34–43.

Borlaf-Mena, I., Badea, O., Tanase, M.A., 2021. Assessing the Utility of Sentinel-1 Coherence Time Series for Temperate and Tropical Forest Mapping. Remote. Sens. 13 (23), 4814. http://dx.doi.org/10.3390/rs13234814.

Brockmann Consult GmbH, 2021. Xcube - An xarray-based EO data cube toolkit — xcube 1.8.0.dev0 documentation. URL: https://xcube.readthedocs.io/en/latest/.

Chatenoux, B., Richard, J.-P., Small, D., Roeoesli, C., Wingate, V., Poussin, C., Rodila, D., Peduzzi, P., Steinmeier, C., Ginzler, C., Psomas, A., Schaepman, M.E., Giuliani, G., 2021. The Swiss data cube, analysis ready data archive using earth observations of Switzerland. Sci. Data 8 (1), 295. http://dx.doi.org/10.1038/s41597-021-01076-6.

Cipoletta, S.R., Sciarra, R., 2024. copernicus Sentinel Data Access Annual Report 2023. URL: https://sentiwiki.copernicus.eu/_attachments/1673407/COPE-SRCO-RP-240521%20-%20Sentinel%20Data%20Access%20Annual%20Report%202023%20-%201.1.pdf?inst-v=86d5ab7c-f08e-4690-a070-6d2a33e3cade.

Craglia, M., Nativi, S., 2018. Mind the Gap: Big Data vs. interoperability and reproducibility of science. Earth Obs. Open Sci. Innov. 121–141.

Crevier, D., Lepage, R., 1997. Knowledge-Based Image Understanding Systems: A Survey. Comput. Vis. Image Underst. 67 (2), 161–185. http://dx.doi.org/10.1006/cviu.1996.0520.

Dask Development Team, 2016. Dask: Library for dynamic task scheduling. URL: http://dask.pydata.org.

Dhu, T., Giuliani, G., Juárez, J., Kavvada, A., Killough, B., Merodio, P., Minchin, S., Ramage, S., 2019. National Open Data Cubes and Their Contribution to Country-Level Development Policies and Practices. Data 4 (4), 144. http://dx.doi.org/10.3390/data4040144.

Di Gregorio, A., Henry, M., Donegan, E., Finegold, Y., Latham, J., Jonckheere, I., Cumani, R., 2016. Land Cover Classification System: Advanced Database Gateway. FAO, Rome, Italy.

Dostálová, A., Wagner, W., Milenković, M., Hollaus, M., 2018. Annual seasonality in Sentinel-1 signal for forest mapping and forest type classification. Int. J. Remote Sens. 39 (21), 7738–7760. http://dx.doi.org/10.1080/01431161.2018.1479788.

Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., Meygret, A., Spoto, F., Sy, O., Marchese, F., Bargellini, P., 2012. Sentinel-2: ESA's Optical High-Resolution Mission for GMES Operational Services. Remote Sens. Environ. 120, 25–36. http://dx.doi.org/10.1016/j.rse.2011.11.026.

Euro Data Cube Consortium, Euro Data Cube. URL: https://eurodatacube.com/.

Ferreira, K.R., Queiroz, G.R., Vinhas, L., Marujo, R.F.B., Simoes, R.E.O., Picoli, M.C.A., Camara, G., Cartaxo, R., Gomes, V.C.F., Santos, L.A., Sanchez, A.H., Arcanjo, J.S., Fronza, J.G., Noronha, C.A., Costa, R.W., Zaglia, M.C., Zioti, F., Korting, T.S., Soares, A.R., Chaves, M.E.D., Fonseca, L.M.G., 2020. Earth Observation Data Cubes for Brazil: Requirements, Methodology and Products. Remote. Sens. 12 (24), http://dx.doi.org/10.3390/rs12244033.

Frantz, D., 2019. FORCE—Landsat + Sentinel-2 Analysis Ready Data and Beyond. Remote. Sens. 11 (9), 1124. http://dx.doi.org/10.3390/rs11091124.

Frolking, S., Palace, M.W., Clark, D.B., Chambers, J.Q., Shugart, H.H., Hurtt, G.C., 2009. Forest disturbance and recovery: A general review in the context of spaceborne remote sensing of impacts on aboveground biomass and canopy structure. J. Geophys. Res.: Biogeosciences 114 (G2), http://dx.doi.org/10.1029/2008JG000911.

Gao, Y., Skutsch, M., Paneque-Gálvez, J., Ghilardi, A., 2020. Remote sensing of forest degradation: a review. Environ. Res. Lett. 15 (10), 103001. http://dx.doi.org/10.1088/1748-9326/abaad7.

Giuliani, G., Chatenoux, B., De Bono, A., Rodila, D., Richard, J.-P., Allenbach, K., Dao, H., Peduzzi, P., 2017. Building an Earth Observations Data Cube: lessons learned from the Swiss Data Cube (SDC) on generating Analysis Ready Data (ARD). Big Earth Data 1 (1–2), 100–117. http://dx.doi.org/10.1080/20964471.2017.1398903.

Giuliani, G., Chatenoux, B., Piller, T., Moser, F., Lacroix, P., 2020. Data Cube on Demand (DCoD): Generating an earth observation Data Cube anywhere in the world. Int. J. Appl. Earth Obs. Geoinf. 87, 102035. http://dx.doi.org/10.1016/j.jag.2019.102035.

Giuliani, G., Masó, J., Mazzetti, P., Nativi, S., Zabala, A., 2019. Paving the Way to Increased Interoperability of Earth Observations Data Cubes. Data 4 (3), 113. http://dx.doi.org/10.3390/data4030113.

Gomes, V., Queiroz, G., Ferreira, K., 2020. An Overview of Platforms for Big Earth Observation Data Management and Analysis. Remote. Sens. 12 (8), 1253. http://dx.doi.org/10.3390/rs12081253.

Goodenough, D., Goldberg, M., Plunkett, G., Zelek, J., 1987. An Expert System for Remote Sensing. IEEE Trans. Geosci. Remote Sens. GE-25 (3), 349–359. http://dx.doi.org/10.1109/TGRS.1987.289805.

Google, 2024. Google blockly - The web-based visual programming editor. Google, URL: https://github.com/google/blockly.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R., 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sens. Environ. 202, 18–27. http://dx.doi.org/10.1016/j.rse.2017.06.031.

Guo, H., Liu, Z., Jiang, H., Wang, C., Liu, J., Liang, D., 2017. Big Earth Data: a new challenge and opportunity for Digital Earth's development. Int. J. Digit. Earth 10 (1), 1–12. http://dx.doi.org/10.1080/17538947.2016.1264490.

Hagedorn, F., Rigling, A., Bebi, P., 2006. Wo Bäume nicht mehr wachsen können: Die Waldgrenze. Die Alp. 9, 52–55.

Hirschmugl, M., Gallaun, H., Dees, M., Datta, P., Deutscher, J., Koutsias, N., Schardt, M., 2017. Methods for Mapping Forest Disturbance and Degradation from Optical Earth Observation Data: a Review. Curr. For. Rep. 3 (1), 32–45. http://dx.doi.org/10.1007/s40725-017-0047-2.

Hoeser, T., Bachofer, F., Kuenzer, C., 2020. Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review - Part II: Applications. Remote. Sens. 12 (18), 3053. http://dx.doi.org/10.3390/rs12183053.

Hoeser, T., Kuenzer, C., 2020. Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review - Part I: Evolution and Recent Trends. Remote. Sens. 12 (10), 1667. http://dx.doi.org/10.3390/rs12101667.

Hoyer, S., Hamman, J., 2017. Xarray: N-d labeled Arrays and Datasets in Python. J. Open Res. Softw. 5 (1), 10. http://dx.doi.org/10.5334/jors.148.

Jacob, A.W., Vicente-Guijalba, F., Lopez-Martinez, C., Lopez-Sanchez, J.M., Litzinger, M., Kristen, H., Mestre-Quereda, A., Ziolkowski, D., Lavalle, M., Notarnicola, C., Suresh, G., Antropov, O., Ge, S., Praks, J., Ban, Y., Pottier, E., Mallorqui Franquet, J.J., Duro, J., Engdahl, M.E., 2020. Sentinel-1 InSAR Coherence for Land Cover Mapping: A Comparison of Multiple Feature-Based Classifiers. IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens. 13, 535–552. http://dx.doi.org/10.1109/JSTARS.2019.2958847.

Kempeneers, P., Soille, P., 2017. Optimizing Sentinel-2 image selection in a Big Data context. Big Earth Data 1 (1–2), 145–158. http://dx.doi.org/10.1080/20964471.2017.1407489.

Killough, B., 2018. Overview of the Open Data Cube Initiative. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, Valencia, pp. 8629–8632. http://dx.doi.org/10.1109/IGARSS.2018.8517694.

Killough, B., Siqueira, A., Dyke, G., 2020. Advancements in the Open Data Cube and Analysis Ready Data — Past, Present and Future. In: IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium. IEEE, Waikoloa, HI, USA, pp. 3373–3375. http://dx.doi.org/10.1109/IGARSS39084.2020.9324712.

Laurini, R., Thompson, D., 1992. Fundamentals of spatial information systems, vol. 37, Academic Press.

Lausch, A., Erasmi, S., King, D., Magdon, P., Heurich, M., 2016. Understanding Forest Health with Remote Sensing - Part I - a Review of Spectral Traits, Processes and Remote-Sensing Characteristics. Remote. Sens. 8 (12), 1029. http://dx.doi.org/10.3390/rs8121029.

Lausch, A., Erasmi, S., King, D., Magdon, P., Heurich, M., 2017. Understanding Forest Health with Remote Sensing - Part II - a Review of Approaches and Data Models. Remote. Sens. 9 (2), 129. http://dx.doi.org/10.3390/rs9020129.

Lewis, A., Lymburner, L., Purss, M.B.J., Brooke, B., Evans, B., Ip, A., Dekker, A.G., Irons, J.R., Minchin, S., Mueller, N., Oliver, S., Roberts, D., Ryan, B., Thankappan, M., Woodcock, R., Wyborn, L., 2016. Rapid, high-resolution detection of environmental change over continental scales from satellite data – the Earth Observation Data Cube. Int. J. Digit. Earth 9 (1), 106–111. http://dx.doi.org/10.1080/17538947.2015.1111952.

Lewis, A., Oliver, S., Lymburner, L., Evans, B., Wyborn, L., Mueller, N., Raevksi, G., Hooke, J., Woodcock, R., Sixsmith, J., Wu, W., Tan, P., Li, F., Killough, B., Minchin, S., Roberts, D., Ayers, D., Bala, B., Dwyer, J., Dekker, A., Dhu, T., Hicks, A., Ip, A., Purss, M., Richards, C., Sagar, S., Trenham, C., Wang, P., Wang, L.-W., 2017. The Australian Geoscience Data Cube — Foundations and lessons learned. Remote Sens. Environ. 202, 276–292. http://dx.doi.org/10.1016/j.rse.2017.03.015.

Mahecha, M.D., Gans, F., Brandt, G., Christiansen, R., Cornell, S.E., Fomferra, N., Kraemer, G., Peters, J., Bodesheim, P., Camps-Valls, G., Donges, J.F., Dorigo, W., Estupinan-Suarez, L.M., Gutierrez-Velez, V.H., Gutwin, M., Jung, M., Londoño, M.C., Miralles, D.G., Papastefanou, P., Reichstein, M., 2020. Earth system data cubes unravel global multivariate dynamics. Earth Syst. Dyn. 11 (1), 201–234. http://dx.doi.org/10.5194/esd-11-201-2020.

Maso, J., Zabala, A., Serral, I., Pons, X., 2019. A Portal Offering Standard Visualization and Analysis on top of an Open Data Cube for Sub-National Regions: The Catalan Data Cube Example. Data 4 (3), 96. http://dx.doi.org/10.3390/data4030096.

Matsuyama, T., 1993. Expert Systems for Image Processing, Analysis, and Recognition: Declarative Knowledge Representation for Computer Vision. In: Advances in Electronics and Electron Physics. 86, Elsevier, pp. 81–171. http://dx.doi.org/10.1016/S0065-2539(08)60154-7.

Matsuyama, T., Hwang, V.S.-S., 1990. SIGMA: A Knowledge-Based Aerial Image Understanding System. Plenum Press, New York, NY, USA; London, UK.

Mazzocchi, F., 2015. Could Big Data be the end of theory in science?: A few remarks on the epistemology of data-driven science. EMBO Rep. 16 (10), 1250–1255. http://dx.doi.org/10.15252/embr.201541001.

Microsoft Open Source, Emanuele, R., Morris, D., Augspurger, T., McFarland, Matt, 2022. Microsoft/PlanetaryComputer: October 2022. http://dx.doi.org/10.5281/zenodo.7261897.

Montero, D., Aybar, C., Ji, C., Kraemer, G., Söchting, M., Teber, K., Mahecha, M.D., 2024a. On-Demand Earth System Data Cubes. In: IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium. pp. 7529–7532. http://dx.doi.org/10.1109/IGARSS53475.2024.10640742.

Montero, D., Kraemer, G., Anghelea, A., Aybar, C., Brandt, G., Camps-Valls, G., Cremer, F., Flik, I., Gans, F., Habershon, S., Ji, C., Kattenborn, T., Martínez-Ferrer, L., Martinuzzi, F., Reinhardt, M., Söchting, M., Teber, K., Mahecha, M.D., 2024b. Earth System Data Cubes: Avenues for advancing Earth system research. http://dx.doi.org/10.48550/arXiv.2408.02348.

Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. ISPRS J. Photogramm. Remote Sens. 66 (3), 247–259. http://dx.doi.org/10.1016/j.isprsjprs.2010.11.001.

Neteler, M., Bowman, M.H., Landa, M., Metz, M., 2012. GRASS GIS: A multi-purpose open source GIS. Environ. Model. Softw. 31, 124–130. http://dx.doi.org/10.1016/j.envsoft.2011.11.014.

Nikaein, T., Iannini, L., Molijn, R.A., Lopez-Dekker, P., 2021. On the Value of Sentinel-1 InSAR Coherence Time-Series for Vegetation Classification. Remote. Sens. 13 (16), 3300. http://dx.doi.org/10.3390/rs13163300.

Paulino, E.R., Schlerf, M., Röder, A., Stoffels, J., Udelhoven, T., 2024. Forest disturbance characterization in the era of earth observation big data: A mapping review. Int. J. Appl. Earth Obs. Geoinf. 128, 103755. http://dx.doi.org/10.1016/j.jag.2024.103755.

Pebesma, E., Bivand, R., 2023. Spatial Data Science: With applications in R. Chapman and Hall/CRC, London, http://dx.doi.org/10.1201/9780429459016.

Peters, J., Neumann, A., Jaeger, M., Gienapp, L., Umlauft, J., 2025. Ml4xcube: Machine learning toolkits for earth system data cubes. In: Proceedings of the AAAI conference on artificial intelligence. 39, pp. 28302–28311. http://dx.doi.org/10.1609/aaai.v39i27.35051.

Quang, N.H., Tuan, V.A., Hao, N.T.P., Hang, L.T.T., Hung, N.M., Anh, V.L., Phuong, L.T.M., Carrie, R., 2019. Synthetic aperture radar and optical remote sensing image fusion for flood monitoring in the Vietnam lower Mekong basin: a prototype application for the Vietnam Open Data Cube. Eur. J. Remote. Sens. 52 (1), 599–612. http://dx.doi.org/10.1080/22797254.2019.1698319.

Rowley, J., 2007. The wisdom hierarchy: representations of the DIKW hierarchy. J. Inf. Sci. 33 (2), 163–180. http://dx.doi.org/10.1177/0165551506070706.

Scheider, S., Ostermann, F.O., Adams, B., 2017. Why good data analysts need to be critical synthesists. Determining the role of semantics in data analysis. Future Gener. Comput. Syst. 72, 11–22. http://dx.doi.org/10.1016/j.future.2017.02.046.

Simoes, R., Camara, G., Queiroz, G., Souza, F., Andrade, P.R., Santos, L., Carvalho, A., Ferreira, K., 2021a. Satellite Image Time Series Analysis for Big Earth Observation Data. Remote. Sens. 13 (13), 2428. http://dx.doi.org/10.3390/rs13132428.

Simoes, R., Souza, F., Zaglia, M., Queiroz, G.R., Santos, R., Ferreira, K., 2021b. Rstac: An R Package to Access Spatiotemporal Asset Catalog Satellite Imagery. In: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. pp. 7674–7677. http://dx.doi.org/10.1109/IGARSS47720.2021.9553518.

Storch, T., Reck, C., Holzwarth, S., Wiegers, B., Mandery, N., Raape, U., Strobl, C., Volkmann, R., Böttcher, M., Hirner, A., Senft, J., Plesia, N., Kukuk, T., Meissl, S., Felske, J.-R., Heege, T., Keuck, V., Schmidt, M., Staudenrausch, H., 2019. Insights into CODE-DE – Germany's Copernicus data and exploitation platform. Big Earth Data 3 (4), 338–361. http://dx.doi.org/10.1080/20964471.2019.1692297.

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. In: Big Data from Space. European Commission Joint Research Centre, Toulouse, France, pp. 32–35. http://dx.doi.org/10.2760/383579, ISSN: 1831-9424.

Sudmanns, M., Augustin, H., Killough, B., Giuliani, G., Tiede, D., Leith, A., Yuan, F., Lewis, A., 2023. Think global, cube local: an Earth Observation Data Cube's contribution to the Digital Earth vision. Big Earth Data 7 (3), 831–859. http://dx.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. Sens. 13 (23), 4807. http://dx.doi.org/10.3390/rs13234807.

Sudmanns, M., Tiede, D., Augustin, H., Lang, S., 2020a. Assessing global Sentinel-2 coverage dynamics and data availability for operational Earth observation (EO) applications using the EO-Compass. Int. J. Digit. Earth 13 (7), 768–784. http://dx.doi.org/10.1080/17538947.2019.1572799.

Sudmanns, M., Tiede, D., Lang, S., Bergstedt, H., Trost, G., Augustin, H., Baraldi, A., Blaschke, T., 2020b. Big Earth data: disruptive changes in Earth observation data management and analysis? Int. J. Digit. Earth 13 (7), 832–850. http://dx.doi.org/10.1080/17538947.2019.1585976.

Van Der Meer, L., Sudmanns, M., Augustin, H., Baraldi, A., Tiede, D., 2022. Semantic Querying in Earth Observation Data Cubes. Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci. XLVIII-4/W1-2022, 503–510. http://dx.doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-503-2022.

Viana-Soto, A., Senf, C., 2025. The European Forest Disturbance Atlas: a forest disturbance monitoring system using the Landsat archive. Earth Syst. Sci. Data 17 (6), 2373–2404. http://dx.doi.org/10.5194/essd-17-2373-2025.

Wagemann, J., Siemen, S., Seeger, B., Bendix, J., 2021. Users of open Big Earth data – An analysis of the current state. Comput. Geosci. 157, 104916. http://dx.doi.org/10.1016/j.cageo.2021.104916.

Wilson, A.M., Jetz, W., 2016. Remotely Sensed High-Resolution Global Cloud Dynamics for Predicting Ecosystem and Biodiversity Distributions. In: Loreau, M. (Ed.), PLOS Biology 14 (3), e1002415. http://dx.doi.org/10.1371/journal.pbio.1002415.

Woodcock, C.E., Allen, R., Anderson, M., Belward, A., Bindschadler, R., Cohen, W., Gao, F., Goward, S.N., Helder, D., Helmer, E., Nemani, R., Oreopoulos, L., Schott, J., Thenkabail, P.S., Vermote, E.F., Vogelmann, J., Wulder, M.A., Wynne, R., 2008. Free Access to Landsat Imagery. Science 320 (5879), http://dx.doi.org/10.1126/science.320.5879.1011a.

Wulder, M.A., Masek, J.G., Cohen, W.B., Loveland, T.R., Woodcock, C.E., 2012. Opening the archive: Howfree data has enabled the science and monitoring promise of Landsat. Remote Sens. Environ. 122, 2–10. http://dx.doi.org/10.1016/j.rse.2012.01.010.

Yuan, F., Lewis, A., Leith, A., Dhar, T., Gavin, D., 2021. Analysis Ready Data for Africa. In: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. IEEE, Brussels, Belgium, pp. 1789–1791. http://dx.doi.org/10.1109/IGARSS47720.2021.9554019.

Zanaga, D., Van De Kerchove, R., De Keersmaecker, W., Souverijns, N., Brockmann, C., Quast, R., Wevers, J., Grosu, A., Paccini, A., Vergnaud, S., Cartus, O., Santoro, M., Fritz, S., Georgieva, I., Lesiv, M., Carter, S., Herold, M., Li, L., Tsendbazar, N.E., Ramoino, F., Arino, O., 2021. ESA WorldCover 10 m 2020 v100. http://dx.doi.org/10.5281/zenodo.5571936.

Zhu, X.X., Tuia, D., Mou, L., Xia, G.S., Zhang, L., Xu, F., Fraundorfer, F., 2017. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. IEEE Geosci. Remote. Sens. Mag. 5 (4), 8–36. http://dx.doi.org/10.1109/MGRS.2017.2762307.

Zhu, Z., Wulder, M.A., Roy, D.P., Woodcock, C.E., Hansen, M.C., Radeloff, V.C., Healey, S.P., Schaaf, C., Hostert, P., Strobl, P., Pekel, J.F., Lymburner, L., Pahlevan, N., Scambos, T.A., 2019. Benefits of the free and open Landsat data policy. Remote Sens. Environ. 224, 382–385. http://dx.doi.org/10.1016/j.rse.2019.02.016.