Contents lists available at ScienceDirect

SoftwareX

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

Original software publication

EODIE — Earth Observation Data Information Extractor

Samantha Wittke[a,b,*], Anne Fouilloux[c], Petteri Lehti[b,d], Juuso Varho[b,d], Arttu Kivimäki[b], Maiju Karhu[b], Mika Karjalainen[b], Matti Vaaja[a], Eetu Puttonen[b]

[a] *Department of Built Environment, Aalto University, Finland*
[b] *Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, National Land Survey of Finland, Finland*
[c] *Department of Geosciences, University of Oslo, Norway*
[d] *Department of Applied Physics, Aalto University, Finland*

ARTICLE INFO

Article history:
Received 8 March 2023
Received in revised form 4 May 2023
Accepted 21 May 2023

Keywords:
Remote sensing
Big data processing
Earth observation
Open-source software

ABSTRACT

Remote sensing satellites provide a vast amount of data to monitor and observe Earth's surface and events on it. To use these data efficiently in subsequent analysis and decision-making, highly automated easy-to-use tools are needed. Here, we present Earth Observation Data Information Extractor (EODIE). EODIE is a toolkit to extract object-level time-series information from several multispectral satellite remote sensing platforms and to produce analysis-ready products for subsequent data analysis. EODIE has a modular design that makes it adjustable for end-user requirements. Users have a possibility to exchange and add modules in EODIE for flexible processing in different computing environments. With EODIE, remote sensing data can be processed to object level array, geotiff or statistics information of different (vegetation) indices or plain wavelength intervals.

Code metadata

Current code version	2.0.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00147
Code Ocean compute capsule	–
Legal Code License	GNU GPL v3.0
Code versioning system used	git
Software code languages, tools, and services used	Python 3
Compilation requirements, operating environments & dependencies	https://raw.githubusercontent.com/samumantha/EODIE/main/environment.yml
If available Link to developer documentation/manual	https://eodie.readthedocs.io/en/latest/
Support email for questions	samantha.wittke@nls.fi

Software metadata

Current software version	2.0.0
Permanent link to executables of this version	https://doi.org/10.5281/zenodo.7467514
Legal Software License	GNU GPL v3.0
Computing platforms/Operating Systems	Unix-like, Windows
Installation requirements & dependencies	https://raw.githubusercontent.com/samumantha/EODIE/main/environment.yml
If available, link to user manual - if formally published include a reference to the publication in the reference list	https://eodie.readthedocs.io/en/latest/index.html
Support email for questions	samantha.wittke@nls.fi

[*] Correspondence to: Vuorimiehentie 5, 02150 Espoo, Finland.
E-mail address: samantha.wittke@nls.fi (Samantha Wittke).

https://doi.org/10.1016/j.softx.2023.101421

1. Motivation and significance

1.1. Spaceborne remote sensing

In remote sensing, natural or anthropogenic phenomena are observed from a distance by a sensor mounted on a platform on the ground, in the air or in space without interfering with the phenomenon in question. With spaceborne platforms, large areas on the Earth can be observed over a long time period with high spatial resolution. Multiple Earth Observation (EO) satellites are orbiting the Earth at different heights, measuring in different wavelengths and producing a large amount of data in different time intervals. The produced time series of objects, areas, or regions help scientists to better understand and monitor processes on Earth's surface [1].

Some EO missions, such as Landsat [2] and Copernicus Sentinel [3] provide their data free of charge for everyone to use. These open EO data are often distributed in tiles to improve data management by allowing focused spatial and temporal queries [4,5].

These data are actively used in many different disciplines such as land use and land cover classification of both natural and urban areas [6–11], precision agriculture [12,13], classification and nitrogen status [12], crop yield estimation and forecasting [14–16], forest fires [17] and vegetation phenology [18,19], estimation of crop growing stages [20], forest composition and its biophysical drivers [21], discriminating plant species [22], estimation of gross primary productivity (GPP) [23], drought detection [24] and impact of mining on vegetation phenology [25].

1.2. Initial motivation

The Earth Observation Data Information Extractor (EODIE) is meant to be a starting point for multidisciplinary researchers from different disciplines wanting to use remote sensing time series data in their studies. The time and resources spent on the actual research question are often limited by time spent on preparing data for research, which is a non-trivial and laborious process. EODIE is designed to help with the EO data preparation step by providing one tool to find the needed data and produce standard form object level output. Thus, EODIE can be integrated as a part of a full data analysis workflow. In addition, it is designed to be flexible in terms of processing environments.

EODIE provides tools to extract object level time series information from multispectral remote sensing datasets. These time series can be on a spectral band (wavelength interval that is recorded by the sensor) level or in the form of (vegetation) indices, such as the Normalized Difference Vegetation Index (NDVI) [26,27] (see appendix Table A.4 for more).

1.3. Other tools

Other software and frameworks for satellite remote sensing data pre-processing, processing and post-processing include, but are not limited to Sentinel Application Platform (SNAP) [28], Framework for Operational Radiometric Correction for Environmental Monitoring (FORCE) [29], Orfeo Toolbox (OTB) [30], QGIS Semi-Automatic Classification Plugin (SCP) [31] as well as the R package Sen2R [32] and sentinelhub for Python [33]. A summary of these tools in comparison to EODIE is shown in Table 1. Other platforms, that include tools for satellite remote sensing data processing include Googles' Earth Engine [34] and Microsofts' Planetary computer.[1]

The goal of EODIE is to provide the user with one tool that combines relevant processing steps from an automated data query to user-defined and analysis-ready end products on object level. EODIE's command line interface (CLI) allows object-level extraction based on vector features and can be executed in one command. EODIE's modular design gives users the freedom to add their own additional processing or analysis modules e.g. to import data from yet unsupported satellite platforms and to compute and export intermediate results, like vegetation indices. Other tools often provide more functionalities that still need to be put together to form a full workflow, as well as limitations towards the computing environment (e.g. graphical interfaces).

1.4. Limitations

EODIE currently supports Sentinel-2 and Landsat 8 platforms as well as generic *geotiff* files. However, only a subset of EODIE functionalities is available for generic *geotiff* input data. Support for other platforms that provide the data and metadata in a similar way to Sentinel-2 or Landsat 8 can be added by providing a platform specific configuration file[2].

EODIE supports the calculation of 17 indices (see Table A.4), which were chosen based on their usage in literature (see for example [19,35–39]) and in the authors' projects. Further indices can be supported with additional functions implemented by the user.

Each raster tile in EODIE is considered spatially and temporally independent, i.e their spatio-temporal calibration is not checked for by default. However, when considering time series over a geographically large areas that are covered with multiple tiles, data from different times and location can only be compared to each other after spatial and temporal calibration. Therefore, the user needs to take the calibration into account before feature extraction. Cloudcover in the area of interest at the time of acquisition is one example for gaps in optical time series data. Filling these gaps with other multispectral, RADAR (Radio Detection and Ranging) or LiDAR (Light Detection and Ranging) data from spaceborne, airborne, or terrestrial remote sensing platforms is an interesting topic for the future.

2. Software description

2.1. Software architecture

EODIE is written in Python [40] and it builds on top of several standard Python packages for efficient numerical and geographic data processing (see Table 2).

Python was chosen due to its popularity in data science and EO. It is efficient for scientific analysis and one of the easiest languages to learn [49]. It also provides a large amount of available packages for functionality extension [49,50] and is reader-friendly, which improves understanding and makes possible code modifications easier for the scientific community.

2.2. Main components

The main components of EODIE are shown in Fig. 1 as purple arrow shapes. As EODIE has modular design, some components can be replaced with data products from other external tools for further customization. An example alteration would be to replace the cloud mask provided by the satellite remote sensing product with cloud masks generated by e.g. Maccs-Atcor Joint Algorithm (MAJA) [51] or FORCE [29] for possibly enhanced cloud-masking, depending on application [52–54].

[1] https://planetarycomputer.microsoft.com/

[2] More information on configuration files can be found in EODIE documentation: https://eodie.readthedocs.io/en/latest/More.html#platform-specific-configuration-files

Table 1
Comparison of EODIE with other tools; Vector overlay: the tool provides the capacity to also read vector files and extract object level information, S2: Sentinel-2, LS: Landsat, CLI: Command Line Interface, GUI: Graphical User Interface.

Tool	Platform	Interfaces	Vector overlay	Indices	Output	Citation
EODIE	S2, LS8, TIF, others with config. files	CLI, GUI	Yes	Many pre-defined (see appendix A.4)	csv, geotiff, array, SQLITE database	–
SCP	ASTER, GOES LS1-8, S1-3 + others	GUI	in QGIS	NDVI, EVI others require own functions	geotiff	[31]
FORCE	LS4-8, S1–2	CLI	No	Many pre-defined	geotiff, envi	[29]
SNAP	S1-S3, RapidEye, SPOT, MODIS, Landsat + others	CLI, GUI	Yes	Many pre-defined	geotiff, csv	[28]
Sen2r	S2	CLI, RStudio	No	Many pre-defined	geotiff, envi	[32]
OTB	Ikonos, S1, Spot 5-7, Worldview 2 + others	CLI, GUI	Yes	Many pre-defined	vector xml, raster	[30]
sentinelhub	Sentinel, Landsat, Modis + others	Python package	Yes	Multiple scripts for calculating indices on Github	JSON, Pandas dataframe	[33]

Table 2
Main dependencies of EODIE (version 2.0.0).

Package	Version	Purpose	Citation
numpy	1.22.4	Efficient numerical array processing	[41]
shapely	1.8.2	Generating geometries from coordinates	[42]
rasterstats	0.17.0	Extraction of pixel values and statistics from raster data	[43]
rasterio	1.2.10	Numpy array extraction from raster input	[44]
gdal	3.5.0	Vector related properties and reprojection	[45]
geopandas	0.11.1	Vector data related pre-processing	[46]
dask	2022.9.0	Process parallelization	[47]
sqlite3	3.39.3	Database handling	[48]

2.2.1. User interface

EODIE is primarily intended to be used via the Command Line Interface (CLI). It is also provided with a graphical web interface on the Galaxy platform (see Section 2.4) and as a Python package.[3]

2.2.2. Workflow

The basic internal workflow of EODIE (for three example tiles) is summarized in Fig. 1. EODIE returns by default object-level outputs and possible intermediate products as defined by the user. It enables the calculation of several vegetation indices and statistics per input polygon from the *vectordata* file. The EODIE CLI has both obligatory and optional command line arguments, as well as configuration flags. For example, flags can define the output formats and whether to include border pixels in statistics calculations. A configuration file in *yaml* format can be used to further customize the processing.

The main inputs to the tool for processing of Sentinel-2 and Landsat 8 data are:

(1) One or multiple **(vegetation) index names** (chosen from Table A.4) to be calculated. Here, also raw bands (as available from satellite remote sensing product) can be chosen.
(2) The **timeframe** defines the time window of interest that will be processed for the available EO data in the source location.
(3) The objects of interest as valid georeferenced polygons provided in supported **vector data** format. Each polygon should have a unique ID which will be used as the identifier for the polygon. The correct geographic reprojection between vector and raster data is handled by EODIE.
(4) The input **raster data** can be a single or multiple raster data products from any of the supported platforms.

Table 3
Example EODIE output default statistics with differently sized polygons; *id* is the unique identifier of each polygon, *count* represents the number of valid (i.e. non-masked) pixels and the rest of the columns represent the mean, standard deviation and median values of a given index of all valid pixels within the polygon.

id	count	mean	std	median
1	735	0.882	0.017	0.885
2	131	0.696	0.148	0.745
3	225	0.451	0.239	0.385
4	439	0.492	0.224	0.493
5	36	0.563	0.066	0.549

The input raster data to be processed are validated and filtered based on the user-defined time frame and geographic location of the polygons of interest. From the filtered dataset, the raster bands and cloud mask are generated, or extracted from the product. The raster bands are used to calculate the indices (Table A.4) requested by the user. Then, the cloud mask is used to mask out invalid pixels (e.g. under cloud, cloud shadow or snow). Finally, the unmasked (i.e. cloud-free) pixels within each polygon are extracted and stored in the user requested format.

EODIE has several different output options: Statistics over each polygon (Table 3) can be extracted directly into a single *SQLite database* [48] file or multiple *comma-separated value* (CSV) files split by date and tile, which can be combined with provided auxiliary scripts. All pixel values per polygon can also be exported into arrays and stored as *pickle* [55] or *geotiff* [56].

[3] https://anaconda.org/conda-forge/eodie

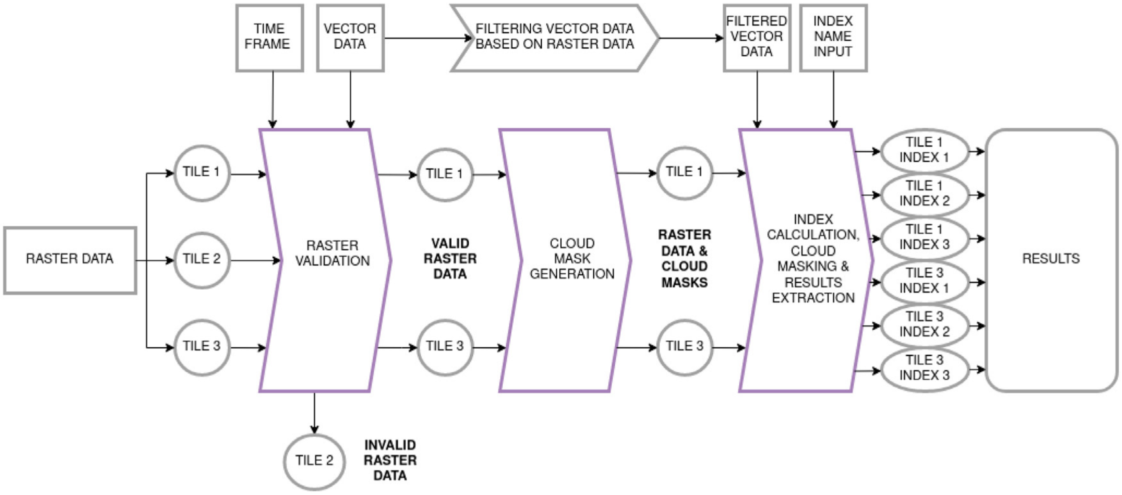
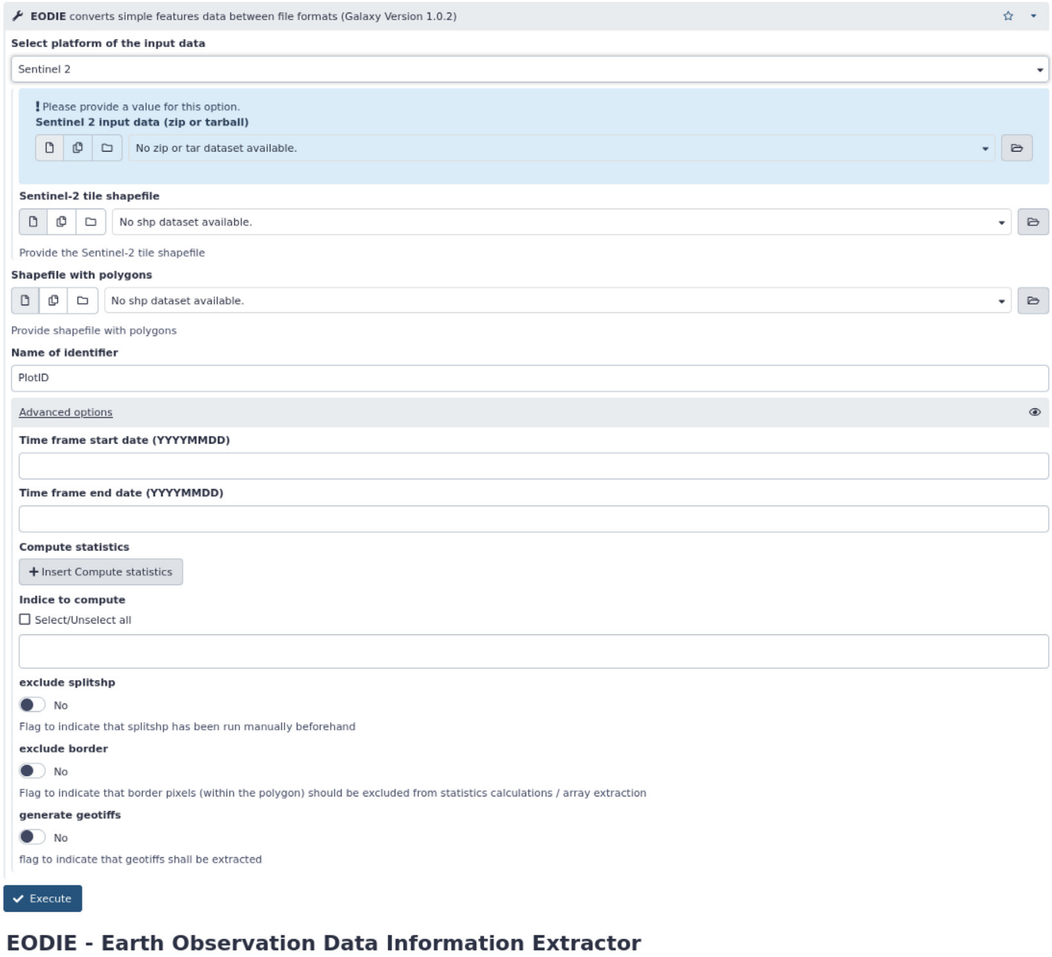


Fig. 1. EODIE workflow, main components are in purple arrow shaped boxes. Rectangular boxes define the input; results formatting are dependent on users choice.



EODIE converts simple features data between file formats (Galaxy Version 1.0.2)

Select platform of the input data

Sentinel 2

! Please provide a value for this option.

Sentinel 2 input data (zip or tarball)

No zip or tar dataset available.

Sentinel-2 tile shapefile

No shp dataset available.

Provide the Sentinel-2 tile shapefile

Shapefile with polygons

No shp dataset available.

Provide shapefile with polygons

Name of identifier

PlotID

Advanced options

Time frame start date (YYYYMMDD)

Time frame end date (YYYYMMDD)

Compute statistics

+ Insert Compute statistics

Indice to compute

□ Select/Unselect all

exclude splitshp

● No

Flag to indicate that splitshp has been run manually beforehand

exclude border

● No

Flag to indicate that border pixels (within the polygon) should be excluded from statistics calculations / array extraction

generate geotiffs

● No

flag to indicate that geotiffs shall be extracted

✓ Execute

EODIE - Earth Observation Data Information Extractor

Fig. 2. Screenshot of the EODIE Graphical User Interface (GUI) on Galaxy platform.

2.2.3. Scalability

EODIE uses *dask* [47], an open-source Python library, for internal parallelization of the process. The three internal EODIE processes parallelized with *dask.delayed* are presented in Fig. 1 in purple colored arrows. The available resources are reported to *dask* and the computations are distributed automatically to make the best use of the resources given. This means that EODIE can be run on different computing infrastructures, from a personal computer to cloud computing systems and computing clusters. The resource usage depends on the amount of raster data that needs to be processed, which is defined by the total area of interest, the requested time frame, and the number of object polygons in the vector data.

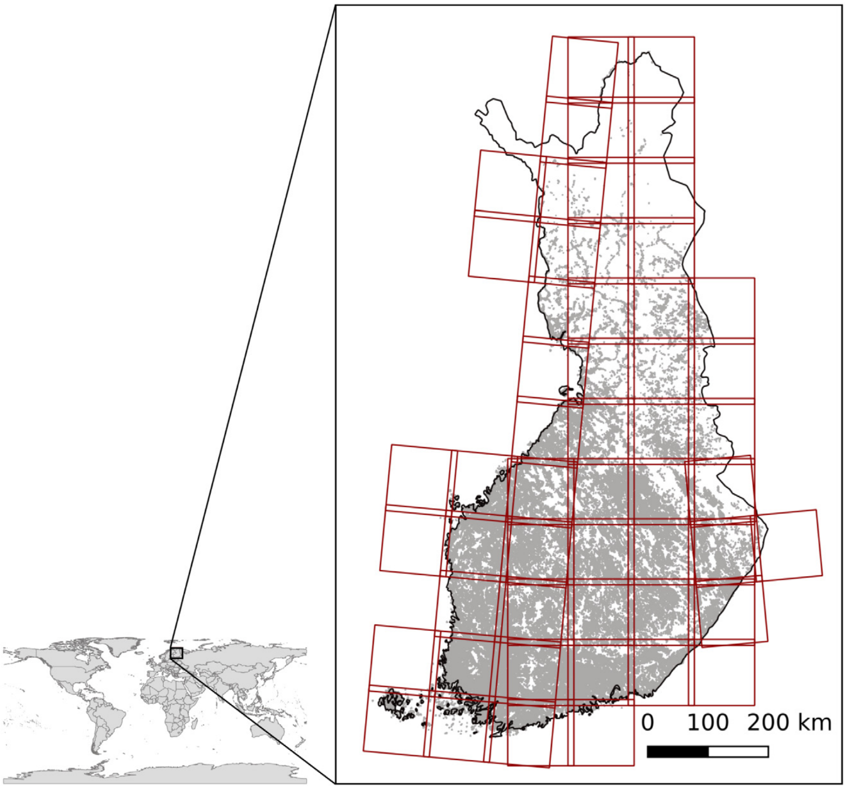


Fig. 3. 63 Sentinel-2 tiles in red covering those parts of Finland, where crops were grown in 2020. All – about one million – agricultural field parcels are shown in grey. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2.3. Software packaging

To facilitate the usage of EODIE on different infrastructures, EODIE[4] has been packaged using Conda [57] and made available on the *conda-forge* channel [58].

2.4. EODIE graphical user interface

EODIE's Graphical User Interface (GUI) is provided on the Galaxy platform[5] to promote findable, accessible, interoperable and reusable (FAIR) research software practices [59]. Galaxy is an open-source workflow management system with both a command line and web-based interface that allows accessible, reproducible, and transparent computational research. The Galaxy Tool for EODIE (available on the European Galaxy instance,[6]; a registered European Open Science Cloud service) offers a GUI to end-users to increase its FAIRness. It makes use of the EODIE Conda package[7], which is provided in a container – available for Docker and Singularity – that can be deployed on any Galaxy instance or used as a standalone fully reproducible tool.

The EODIE Galaxy tool wrapper hides the complexity and command line look of the tool and guides users in the usage. Fig. 2 shows the current EODIE GUI with two sections:

1. The first section concerns mandatory EODIE software input parameters such as platform (Sentinel-2, Landsat 8 or generic *geotiff*), location of input data and the area and polygons of interest as well as the name of unique polygon identifiers;
2. The second section provides "advanced options" such as time range, selection of statistics (mean, minimum, maximum, etc.) and further (vegetation) indices to compute.

EODIE documentation also provides a tutorial[8] on the use of the Galaxy platform tool.

3. Illustrative example

A typical use case for EODIE is the provision of different vegetation index time series of agricultural field parcels over the growing season from Sentinel-2 data. In this example, we utilized the freely available Finnish field parcel information of 2020 [60] provided by the Finnish Food Authority as a *GeoPackage*. The vector dataset includes all – about one million – field parcels in Finland that cover in total an area of 2.3 million hectares all over Finland (see also Fig. 3). The dataset is annually updated and contains information on the crop species and location, shape and size of the field parcels. The crop species field provides information if a field is planted with a single species or a mixture of several species, is fallow, or in special use.

The input raster data for this example is Sentinel-2 Bottom-of-Atmosphere (L2A) product over Finland covering the growing seasons of 2019, 2020 and 2021 (see Fig. 3).

To start the process, EODIE is set up to run with the following inputs:

- **–rasterdir /location/of/Sentinel-2data** the path to the directory where the Sentinel-2 data are stored
- **–platform s2** the platform is set to Sentinel-2
- **–vector /location/of/vectordatafile.gpkg** the path to the *vectorfile* to process

[4] EODIE on conda-forge: https://anaconda.org/conda-forge/eodie

[5] https://usegalaxy.eu/

[6] https://usegalaxy.eu/root?tool_id=toolshed.g2.bx.psu.edu/repos/climate/eodie/eodie/1.0.2

[7] https://anaconda.org/conda-forge/eodie

[8] https://eodie.readthedocs.io/en/latest/Galaxy.html

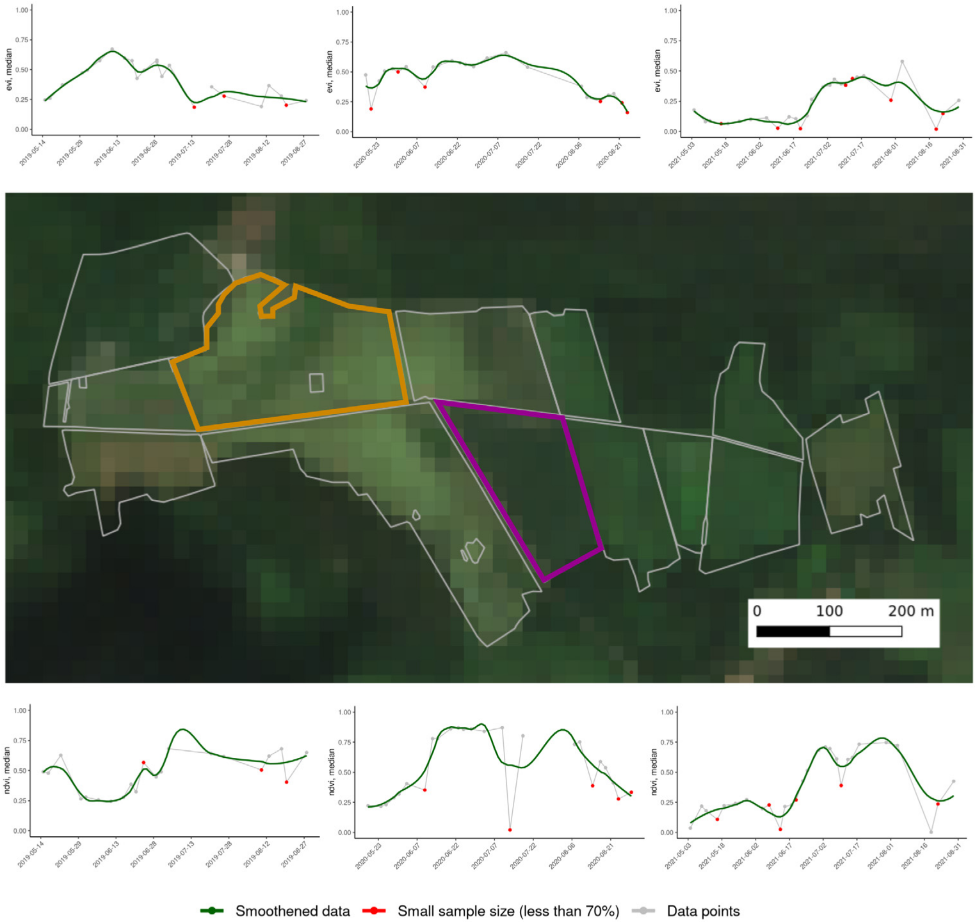


Fig. 4. Example time series plots for EVI (orange field parcel, upper three plots growing grass, barley, rapeseed in the years 2019, 2020 and 2021 respectively from left to right) and NDVI (purple field parcel, bottom three plots growing oats, spring wheat, rapeseed in the years 2019, 2020 and 2021 respectively from left to right) extracted from field parcels each growing different crop type in South-Western Finland. Field parcel boundaries of the area are marked with grey outlines and overlaid on the Sentinel-2 true color image from 18 July 2020 in the background. Exact location of the field plots cannot be disclosed due to privacy agreement. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

- **–out /location/of/outputdirectory** the location of the directory where output files shall be stored
- **–id uID** the name of the *vector file* field containing a unique identifier for the polygons, in this case it is a separately created field called "uID"
- **–database_out** flag setting the output format to database
- **–statistics mean median std** statistics to be calculated for each polygon
- **–index ndvi evi ndmi kndvi B08 B04** indices and bands to be calculated and/or extracted for illustration purposes.

Within the process, *dask.delayed* will distribute the workload on the available processing resources.

After all index calculation and extraction processes are completed, the results can be found in the database file. The results can be visualized for example as time series graph which can be seen in Fig. 4.

The total area in the example for Finland is covered by 63 Sentinel-2 tiles for each of the three years. Thanks to *dask*, EODIE can make use of however many computing resources are available on a single laptop or computing cluster. The example presented here was run on supercomputer Puhti at CSC - IT Center for Science.[9]

4. Impact

Based on the authors' experience in multidisciplinary research projects (see B), the process of information extraction from satellite remote sensing time series data is among the most time consuming parts of work. EODIE has been designed to help scientists without extensive remote sensing expertise to preprocess their EO data into analysis-ready format. The tool provides automated information extraction with the possibility for flexible processing workflow adjustments.

EODIE has already been used in multiple agricultural studies [61,62] and its earlier version in one forest study [63] and for crop yield prediction [16]. When these studies were initiated in 2017–2018, the other available tools did not provide the processing steps and flexibility needed to accomplish the tasks in a single step. In addition, EODIE and experiences in developing it have enabled the authors to participate in several national and international research and development projects (see Appendix B).

In the example use case we demonstrated how EODIE can be applied over a large geographical area to extract information for an agricultural case study. EODIE can also be used in a similar fashion in other disciplines, such as forestry, water area monitoring, and urban applications. While the earlier studies using EODIE have focused in Finland, it can be directly used with all globally collected Sentinel-2 and Landsat 8 data without source code modifications.

[9] https://research.csc.fi/-/puhti

Table A.4
(Vegetation) indices available in EODIE.

Index name	Abbreviation	Formula	Reference
Normalized Difference Vegetation Index	NDVI	(NIR - RED)/(NIR + RED)	[26]
Ratio Vegetation Index	RVI	NIR/RED	[64]
Soil-Adjusted Vegetation Index	SAVI	(1.5 * (NIR - RED))/(NIR + RED + 0.5)	[65]
Normalized Burn Ratio	NBR	(NIR - SWIR2)/(NIR + SWIR2)	[66]
Kernel Normalized Difference Vegetation Index	kNDVI	$(1-(-(NIR-RED)^2/(2*(0.5*(NIR+RED)^2)))^2)/(1+(-(NIR-RED)^2/(2*(0.5*(NIR+RED)^2)))^2)$	[67]
Normalized Difference Moisture Index	NDMI	(NIR - SWIR1)/(NIR + SWIR1)	[68]
Normalized Difference Water Index	NDWI	(GREEN - NIR)/(GREEN + NIR)	[69]
Modified Normalized Difference Water Index	MNDWI	(GREEN - SWIR1)/(GREEN + SWIR1)	[70]
Enhanced Vegetation Index	EVI	2.5 * ((NIR - RED)/(NIR + 6 * RED - 7.5 * BLUE + 1))	[71]
Enhanced Vegetation Index 2	EVI2	2.5 * ((NIR - RED)/ (2.4 * RED + NIR + 1))	[72]
Difference Vegetation Index	DVI	NIR - RED	[73]
Chlorophyll Vegetation Index	CVI	(NIR * RED)/(GREEN * GREEN)	[74]
Modified Chlorophyll Absorption in Reflectance Index	MCARI	(R_EDGE - RED - 0.2 * (R_EDGE - GREEN) * (R_EDGE/RED)	[75]
Normalized Difference Index 45	NDI45	(R_EDGE - RED)/(R_EDGE + RED)	[76]
Tasseled Cap (for Sentinel-2)	TCT	Coefficients * [BLUE, GREEN, RED, NIR, SWIR1, SWIR2]	[77]

5. Conclusions

EODIE is a toolkit that automates statistical time series data extraction process from remote sensing data from user-defined areas of interest. At the time of development, no other tool provided the full time series extraction process and other tools could not easily be integrated in the full data analysis workflow in Python.

Our contribution with EODIE is to benefit research and development in many different disciplines by enabling faster prototyping and testing of new concepts with remote sensing time series. With this, we are positive in that EODIE has the potential to foster scientific discoveries and benefit applications in various disciplines to increase the understanding of processes on Earth's surface over time.

While EODIE is fully functional in its current state presented here, there is always room for improvements. Ideas for future work can be found in the EODIE Gitlab Repository issues.[10] EODIE is publicly available with an open license and its functionality extensions and bug fixes can be contributed by everyone.

CRediT authorship contribution statement

Samantha Wittke: Software, Validation, Writing – original draft & editing, Conceptualization, Supervision. **Anne Fouilloux:** Software (Galaxy tool), Writing – original draft & editing. **Petteri Lehti:** Software, Writing - review. **Juuso Varho:** Software, Writing - review. **Arttu Kivimäki:** Software, Writing – review & editing. **Maiju Karhu:** Writing – review & editing. **Mika Karjalainen:** Supervision, Funding acquisition, Writing – review. **Matti Vaaja:** Resources, Supervision, Funding acquisition, Writing – review. **Eetu Puttonen:** Resources, Funding acquisition, Writing – review & editing, Supervision, Conceptualization.

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.

Data availability

Links to used data are provided in the manuscript (Illustrative example section).

Acknowledgments

This project was initiated under the Academy of Finland research project 295047 in collaboration with Paula Litkey and Miloš Pandžić and has been supported also from the project 316096/320075. Part of the work has also been done under the umbrella of Academy of Finland flagship project UNITE (337656). Ms. Wittke acknowledges the PhD grant from Aalto School of Engineering. The authors wish to thank CSC – IT Center for Science and the Open Geospatial Information Infrastructure for Research (oGIIR, urn:nbn:fi:research-infras-2016072513) for generous computational resources and guidance. Maria Yli-Heikkilä from Natural Resources Finland (LUKE) is acknowledged for providing Sentinel-2 data on CSC object storage for the example use case as a part of the Eurostat project "Development of pre-harvest crop yield forecasts with satellite remote sensing data" (EC grant agreement No 831735). We are also grateful for the constructive comments on the documentation by the Nordic-RSE community (Richard Darst, Radovan Bast, Luca Ferranti, Enrico Glerean and

[10] https://gitlab.com/fgi_nls/public/EODIE/-/issues

Matthew West) and valuable comments on the manuscript by Katri Tegel. The development of the EODIE Galaxy Tool has been supported by EOSC-Nordic, a project funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 857652.

Appendix A. Vegetation indices

See Table A.4.

Appendix B. Projects

EODIE or its derivates have enabled or have been used in following projects:

- A project related to deforestation monitoring with Sentinel-1 funded by the European Space Agency.[11]
- Two projects related to crop and crop yield monitoring funded by Eurostat: CROPYIELD.[12] and BIGDATA&EO[13]
- A Business Finland co-creation project, working with Finnish companies in EO-business to find business opportunities (EODIE: 5332/31/2018),
- Project AICropPro[14] investigating machine learning methods combined with crop simulation models funded by the Academy of Finland.
- Multiple larger national agriculture related projects such as DIGITALIS,[15] Peltopiste[16] and Ikivihreä.[17]

References

[1] Lillesand T, Kiefer R, Chipman J. Remote sensing and image interpretation. Wiley; 2015.

[2] Wulder MA, Loveland TR, Roy DP, Crawford CJ, Masek JG, Woodcock CE, et al. Current status of Landsat program, science, and applications. Remote Sens Environ 2019;225:127–47. http://dx.doi.org/10.1016/j.rse.2019.02.015.

[3] European Space Agency. Sentinel-2 user handbook. 2015, URL https://sentinels.copernicus.eu/documents/247904/685211/Sentinel-2_User_Handbook.pdf.

[4] USGS. Landsat WRS 2 scene boundaries KML file. 2022, URL https://www.usgs.gov/media/files/landsat-wrs-2-scene-boundaries-kml-file.

[5] ESA. Data products. 2022, URL https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-2/data-products.

[6] E. D. Chaves M, C. A. Picoli M, D. Sanches I. Recent applications of Landsat 8/OLI and Sentinel-2/MSI for land use and land cover mapping: A systematic review. Remote Sens 2020;12(18). URL https://www.mdpi.com/2072-4292/12/18/3062.

[7] Mohajane M, Essahlaoui A, Oudija F, Hafyani ME, Hmaidi AE, Ouali AE, et al. Land use/land cover (LULC) using landsat data series (MSS, TM, ETM and OLI) in Azrou forest, in the Central Middle Atlas of Morocco. Environments 2018;5(12):131. http://dx.doi.org/10.3390/environments5120131.

[8] Mahdianpari M, Salehi B, Rezaee M, Mohammadimanesh F, Zhang Y. Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. Remote Sens 2018;10(7). http://dx.doi.org/10.3390/rs10071119.

[9] Tong X-Y, Xia G-S, Lu Q, Shen H, Li S, You S, et al. Land-cover classification with high-resolution remote sensing images using transferable deep models. Remote Sens Environ 2020;237. http://dx.doi.org/10.1016/j.rse.2019.111322.

[10] Hu B, Xu Y, Huang X, Cheng Q, Ding Q, Bai L, et al. Improving urban land cover classification with combined use of Sentinel-2 and Sentinel-1 imagery. ISPRS Int J Geo-Inf 2021;10(8). http://dx.doi.org/10.3390/ijgi10080533.

[11] Zhang T, Su J, Xu Z, Luo Y, Li J. Sentinel-2 satellite imagery for urban land cover classification by optimized random forest classifier. Appl Sci 2021;11(2). http://dx.doi.org/10.3390/app11020543, URL https://www.mdpi.com/2076-3417/11/2/543.

[12] Segarra J, Buchaillot ML, Araus JL, Kefauver SC. Remote sensing for precision agriculture: Sentinel-2 improved features and applications. Agronomy 2020;10(5). URL https://www.mdpi.com/2073-4395/10/5/641.

[13] Duarte L, Teodoro AC, Sousa JJ, Pádua L. QVigourMap: A GIS open source application for the creation of canopy vigour maps. Agronomy 2021;11(5). http://dx.doi.org/10.3390/agronomy11050952, URL https://www.mdpi.com/2073-4395/11/5/952.

[14] Bolton DK, Friedl MA. Forecasting crop yield using remotely sensed vegetation indices and crop phenology metrics. Agricult Forest Meteorol 2013;173:74–84. http://dx.doi.org/10.1016/j.agrformet.2013.01.007.

[15] Nazir A, Ullah S, Saqib ZA, Abbas A, Ali A, Iqbal MS, et al. Estimation and forecasting of rice yield using phenology-based algorithm and linear regression model on Sentinel-II satellite data. Agriculture-Basel 2021;11(10). http://dx.doi.org/10.3390/agriculture11101026.

[16] Yli-Heikkila M, Wittke S, Luotamo M, Puttonen E, Sulkava M, Pellikka P, et al. Scalable crop yield prediction with Sentinel-2 time series and temporal convolutional network. Remote Sens 2022;14(17). http://dx.doi.org/10.3390/rs14174193, URL https://www.mdpi.com/2072-4292/14/17/4193.

[17] Sandamali KUJ, Chathuranga KAM. Quantification of burned severity of the forest fire using Sentinel-2 remote sensing images: A case study in the Ella Sri Lanka. Res Rev: J Environ Sci 2021;3(2):1–12.

[18] Duarte L, Teodoro AC, Gonçalves H. Deriving phenological metrics from NDVI through an open source tool developed in QGIS. In: Michel U, Schulz K, editors. Earth resources and environmental remote sensing/GIS applications V, vol. 9245. SPIE, International Society for Optics and Photonics; 2014, 924511. http://dx.doi.org/10.1117/12.2066136.

[19] Misra G, Cawkwell F, Wingler A. Status of phenological research using Sentinel-2 data: A review. Remote Sens 2020;12(17). URL https://www.mdpi.com/2072-4292/12/17/2760.

[20] Diao C. Remote sensing phenological monitoring framework to characterize corn and soybean physiological growing stages. Remote Sens Environ 2020;248. http://dx.doi.org/10.1016/j.rse.2020.111960.

[21] Bajocco S, Ferrara C, Alivernini A, Bascietto M, Ricotta C. Remotely-sensed phenology of Italian forests: Going beyond the species. Int J Appl Earth Obs Geoinf 2019;74:314–21. http://dx.doi.org/10.1016/j.jag.2018.10.003.

[22] Madonsela S, Cho MA, Mathieu R, Mutanga O, Ramoelo A, Kaszta Z, et al. Multi-phenology WorldView-2 imagery improves remote sensing of savannah tree species. Int J Appl Earth Obs Geoinf 2017;58:65–73. http://dx.doi.org/10.1016/j.jag.2017.01.018.

[23] Junttila S, Kljun N, Eklundh L. Comparison of light use efficiency, plant phenology index, and light response function-based GPP models in the northern forest landscape. In: IEEE international geoscience and remote sensing symposium IGARSS. 2021, p. 6917–20. http://dx.doi.org/10.1109/IGARSS47720.2021.9554177.

[24] Li L, Qiu B, Guo W, Zhang Y, Song Q, Chen J. Phenological and physiological responses of the terrestrial ecosystem to the 2019 drought event in Southwest China: Insights from satellite measurements and the SSiB2 model. Int J Appl Earth Obs Geoinf 2022;111. http://dx.doi.org/10.1016/j.jag.2022.102832.

[25] Sun X, Yuan L, Liu M, Liang S, Li D, Liu L. Quantitative estimation for the impact of mining activities on vegetation phenology and identifying its controlling factors from Sentinel-2 time series. Int J Appl Earth Obs Geoinf 2022;111. http://dx.doi.org/10.1016/j.jag.2022.102814.

[26] Rouse JJ, Haas R, Schell J, Deering D. Monitoring vegetation systems in the Great Plains with ERTS. In: Proceedings of the third earth resources technology satellite-1 symposium, Washington, DC, USA. 1973.

[27] Tucker C, Miller L, Pearson R. Measurement of the combined effect of green biomass, chlorophyll, and leaf water on canopy spectroreflectance of the shortgrass prairie. Remote Sens Earth Resour 1973.

[28] SNAP - ESA Sentinel Application Platform, version 8.0.0. 2020, URL http://step.esa.int.

[29] Frantz D. FORCE - Landsat + Sentinel-2 analysis ready data and beyond. Remote Sens 2019;11(9). http://dx.doi.org/10.3390/rs11091124, URL https://www.mdpi.com/2072-4292/11/9/1124.

[30] Grizonnet M, Michel J, Poughon V, Inglada J, Savinaud M, Cresson R. Orfeo ToolBox: Open source processing of remote sensing images. Open Geospatial Data, Softw Stand 2017;2(15):1–8. http://dx.doi.org/10.1186/s40965-017-0031-6.

[31] Congedo L. Semi-Automatic Classification Plugin: A Python tool for the download and processing of remote sensing images in QGIS. J Open Sour Softw 2021;6(64):3172. http://dx.doi.org/10.21105/joss.03172.

[32] Ranghetti L, Boschetti M, Nutini F, Busetto L. "sen2r": An R toolbox for automatically downloading and preprocessing Sentinel-2 satellite data. Comput Geosci 2020;139. http://dx.doi.org/10.1016/j.cageo.2020.104473, URL https://www.sciencedirect.com/science/article/pii/S0098300419304893.

[33] Sentinelhub for Python, Sinergise Ltd. 2022, URL https://sentinelhub-py.readthedocs.io/en/latest/.

[11] http://project.gisat.cz/s14scienceAmazonas/

[12] https://www.luke.fi/en/projects/cropyield

[13] https://www.luke.fi/en/projects/bigdataeo

[14] https://www.luke.fi/en/projects/aicroppro

[15] https://www.luke.fi/en/projects/digitalis-01

[16] https://www.luke.fi/en/projects/peltopist

[17] https://www.luke.fi/en/projects/ikivihrea

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

[35] Xue J, Su B. Significant remote sensing vegetation indices: A review of developments and applications. In: Li C, editor. J Sensors 2017;2017:1353691. http://dx.doi.org/10.1155/2017/1353691, Publisher: Hindawi.

[36] Zeng Y, Hao D, Huete A, Dechant B, Berry J, Chen JM, et al. Optical vegetation indices for monitoring terrestrial ecosystems globally. Nat Rev Earth Environ 2022;3(7):477–93. http://dx.doi.org/10.1038/s43017-022-00298-5.

[37] Bolton DK, Gray JM, Melaas EK, Moon M, Eklundh L, Friedl MA. Continental-scale land surface phenology from harmonized Landsat 8 and Sentinel-2 imagery. Remote Sens Environ 2020;240:111685. http://dx.doi.org/10.1016/j.rse.2020.111685.

[38] Al-Gaadi KA, Hassaballa AA, Tola E, Kayad AG, Madugundu R, Alblewi B, et al. Prediction of potato crop yield using precision agriculture techniques. PLoS One 2016;11(9):e0162219. http://dx.doi.org/10.1371/journal.pone.0162219.

[39] Mutanga O, Adam E, Cho MA. High density biomass estimation for wetland vegetation using WorldView-2 imagery and random forest regression algorithm. Int J Appl Earth Obs Geoinf 2012;18:399–406. http://dx.doi.org/10.1016/j.jag.2012.03.012.

[40] Van Rossum G, Drake FL. Python 3 reference manual. Scotts Valley, CA: CreateSpace; 2009.

[41] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature 2020;585(7825):357–62. http://dx.doi.org/10.1038/s41586-020-2649-2.

[42] Gillies S, et al. Shapely: Manipulation and analysis of geometric objects. 2007, toblerity.org, URL https://github.com/Toblerity/Shapely.

[43] Perry MT. Rasterstats documentation. 2015, URL https://pythonhosted.org/rasterstats/, [Accessed on 18 October 2021].

[44] Gillies S, et al. Rasterio: Geospatial raster I/O for Python programmers. 2013, Mapbox, URL https://github.com/mapbox/rasterio, [Accessed on 18 October 2021.

[45] GDAL/OGR contributors. GDAL/OGR geospatial data abstraction software library. 2021, Open Source Geospatial Foundation, URL https://gdal.org.

[46] Jordahl K, den Bossche JV, Fleischmann M, Wasserman J, McBride J, Gerard J, et al. Geopandas/geopandas: v0.11.1. 2022, http://dx.doi.org/10.5281/zenodo.6894736, Zenodo.

[47] Dask Development Team. Dask: Library for dynamic task scheduling. 2016, URL https://dask.org.

[48] Python Software Foundation. sqlite3 - DB-API 2.0 interface for SQLite databases. 2022, URL https://docs.python.org/3/library/sqlite3.html.

[49] Ayer VM, Miguez S, Toby BH. Why scientists should learn to program in Python. Powder Diffr 2014;29(S2):S48–64. http://dx.doi.org/10.1017/S0885715614000931.

[50] Peng Y, Zhang Y, Hu M. An empirical study for common language features used in Python projects. In: 2021 IEEE international conference on software analysis, evolution and reengineering. SANER, 2021, p. 24–35. http://dx.doi.org/10.1109/SANER50967.2021.00012.

[51] Hagolle O, Huc M, Desjardins C, Auer S, Richter R. MAJA Algorithm Theoretical Basis Document. Zenodo; 2017, http://dx.doi.org/10.5281/zenodo.1209633.

[52] Foga S, Scaramuzza PL, Guo S, Zhu Z, Dilley RD, Beckmann T, et al. Cloud detection algorithm comparison and validation for operational Landsat data products. Remote Sens Environ 2017;194:379–90. http://dx.doi.org/10.1016/j.rse.2017.03.026, URL https://linkinghub.elsevier.com/retrieve/pii/S0034425717301293.

[53] Tarrio K, Tang X, Masek JG, Claverie M, Ju J, Qiu S, et al. Comparison of cloud detection algorithms for Sentinel-2 imagery. Sci Remote Sens 2020;2:100010. http://dx.doi.org/10.1016/j.srs.2020.100010, URL https://www.sciencedirect.com/science/article/pii/S2666017220300092.

[54] Zekoll V, Main-Knorn M, Alonso K, Louis J, Frantz D, Richter R, et al. Comparison of masking algorithms for Sentinel-2 imagery. Remote Sens 2021;13(1):137. http://dx.doi.org/10.3390/rs13010137, URL https://www.mdpi.com/2072-4292/13/1/137.

[55] Python Software Foundation. pickle - Python object serialization. 2022, URL https://docs.python.org/3/library/pickle.html.

[56] Ritter N, Ruth M. The GeoTiff data interchange standard for raster geographic images. Int J Remote Sens 1997;18(7):1637–47. http://dx.doi.org/10.1080/014311697218340.

[57] Conda license. 2017, Continuum Analytics, Inc., URL https://docs.conda.io/projects/conda/en/latest/.

[58] The conda-forge Project: Community-based software distribution built on the conda package format and ecosystem. conda-forge community; 2015, URL https://zenodo.org/record/4774217.

[59] Hong NPC, Katz DS, Barker M, Lamprecht A-L, Martinez C, Psomopoulos FE, et al. FAIR principles for research software (FAIR4RS principles). Research Data Alliance; 2021, http://dx.doi.org/10.15497/RDA00065.

[60] Agency for rural affairs in Finland. Agricultural parcels 2020, 1:5 000. 2022, CSC – IT Center for Science, http://urn.fi/urn:nbn:fi:att:819a0a28-c603-4403-af88-0ca32d5188aa.

[61] Peltonen-Sainio P, Jauhiainen L, Laurila H, Sorvali J, Honkavaara E, Wittke S, Karjalainen M, et al. Land use optimization tool for sustainable intensification of high-latitude agricultural systems. Land Use Policy 2019;88:104104. http://dx.doi.org/10.1016/j.landusepol.2019.104104, URL https://www.sciencedirect.com/science/article/pii/S0264837718319781.

[62] Peltonen-Sainio P, Jauhiainen L, Honkavaara E, Wittke S, Karjalainen M, Puttonen E. Pre-crop values from satellite images for various previous and subsequent crop combinations. Front Plant Sci 2019;10:462. http://dx.doi.org/10.3389/fpls.2019.00462, URL https://www.frontiersin.org/article/10.3389/fpls.2019.00462.

[63] Wittke S, Yu X, Karjalainen M, Hyyppä J, Puttonen E. Comparison of two-dimensional multitemporal Sentinel-2 data with three-dimensional remote sensing data sources for forest inventory parameter estimation over a boreal forest. Int J Appl Earth Obs Geoinf 2019;76:167–78. http://dx.doi.org/10.1016/j.jag.2018.11.009, URL https://www.sciencedirect.com/science/article/pii/S0303243418309462.

[64] Jordan CF. Derivation of leaf-area index from quality of light on the forest floor. Ecology 1969;50(4):663–6. http://dx.doi.org/10.2307/1936256.

[65] Huete A. A soil-adjusted vegetation index (SAVI). Remote Sens Environ 1988;25(3):295–309. http://dx.doi.org/10.1016/0034-4257(88)90106-x.

[66] García ML, Caselles V. Mapping burns and natural reforestation using thematic Mapper data. Geocarto Int 1991;6(1):31–7. http://dx.doi.org/10.1080/10106049109354290.

[67] Camps-Valls G, Campos-Taberner M, Moreno-Martínez Á, Walther S, Duveiller G, Cescatti A, et al. A unified vegetation index for quantifying the terrestrial biosphere. Sci Adv 2021;7(9). http://dx.doi.org/10.1126/sciadv.abc7447.

[68] Gao B. NDWI–A normalized difference water index for remote sensing of vegetation liquid water from space. Remote Sens Environ 1996;58(3):257–66. http://dx.doi.org/10.1016/s0034-4257(96)00067-3.

[69] McFeeters SK. The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. Int J Remote Sens 1996;17(7):1425–32. http://dx.doi.org/10.1080/01431169608948714.

[70] Xu H. Modification of Normalised Difference Water Index (NDWI) to enhance open water features in remotely sensed imagery. Int J Remote Sens 2006;27(14):3025–33. http://dx.doi.org/10.1080/01431160600589179.

[71] Liu HQ, Huete A. A feedback based modification of the NDVI to minimize canopy background and atmospheric noise. IEEE Trans Geosci Remote Sens 1995;33(2):457–65. http://dx.doi.org/10.1109/TGRS.1995.8746027.

[72] Jiang Z, Huete A, Didan K, Miura T. Development of a two-band enhanced vegetation index without a blue band. Remote Sens Environ 2008;112(10):3833–45. http://dx.doi.org/10.1016/j.rse.2008.06.006.

[73] Tucker CJ. Red and photographic infrared linear combinations for monitoring vegetation. Remote Sens Environ 1979;8(2):127–50. http://dx.doi.org/10.1016/0034-4257(79)90013-0.

[74] Vincini M, Frazzi E, D'Alessio P. A broad-band leaf chlorophyll vegetation index at the canopy scale. Precis Agric 2008;9(5):303–19. http://dx.doi.org/10.1007/s11119-008-9075-z.

[75] Daughtry C, Walthall C, Kim M, de Colstoun EB, McMurtrey J. Estimating corn leaf chlorophyll concentration from leaf and canopy reflectance. Remote Sens Environ 2000;74(2):229–39. http://dx.doi.org/10.1016/s0034-4257(00)00113-9.

[76] Delegido J, Verrelst J, Alonso L, Moreno J. Evaluation of Sentinel-2 red-edge bands for empirical estimation of green LAI and chlorophyll content. Sensors 2011;11(7):7063–81. http://dx.doi.org/10.3390/s110707063.

[77] Shi T, Xu H. Derivation of Tasseled cap transformation coefficients for Sentinel-2 MSI at-sensor reflectance data. IEEE J Sel Top Appl Earth Observ Remote Sens 2019;12(10):4038–48. http://dx.doi.org/10.1109/jstars.2019.2938388.