This article has been accepted for publication in IEEE Geoscience and Remote Sensing Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LGRS.2026.3720762

Efficient semantic content-based image retrieval with cloud-optimized raster formats

Luke McQuade
Department of Geoinformatics – Z_GIS
Paris Lodron University Salzburg
Salzburg, Austria
luke.mcquade@plus.ac.at

Martin Sudmanns
Department of Geoinformatics – Z_GIS
Paris Lodron University Salzburg
Salzburg, Austria
martin.sudmanns@plus.ac.at

Dirk Tiede
Department of Geoinformatics – Z_GIS
Paris Lodron University Salzburg
Salzburg, Austria
dirk.tiede@plus.ac.at

***Abstract*—Semantic queries for content-based retrieval of Earth observation imagery, such as "Find the images with less than 5% cloud cover in this area between two dates", rely on aggregating some kind of scene classification data. Given the millions of pixels in a typical image, this can be resource-intensive. If knowledge is required of only the relative distribution of classes, though, do we need to process every pixel? The size and shape (morphology) of natural features, when observed by high-resolution optical sensors such as Sentinel-2, mean that simple downsampling can be used to drastically reduce the processing required for such queries, with only small losses in accuracy. Given an error tolerance of 1%, memory usage can be reduced by 625× and query runtime reduced substantially, for areas of interest of 60 km × 60 km. Using cloud-native technologies such as Cloud Optimized GeoTIFF (COG), this can also lead to significant reductions in network and disk usage. However, systems where these are typically employed make the performance improvements difficult to precisely quantify, due to compounding latencies.**

***Index Terms*—scene classification, land cover, Sentinel-2, cloud-native geospatial**

I. Introduction

Semantic queries in large satellite Earth Observation (EO) datasets with use cases such as "Find the images with less than 5% cloud cover between two dates", or "Find the first image of the year where the vegetation content of this area rises above 20%" would be valuable to search image archives, but are usually not implemented in EO data archives. These require some kind of scene classification information, which is then aggregated to the user's area of interest. Given the millions of pixels in a typical image, and dozens to hundreds of images to query against, this can be resource-intensive. The natural features that underlie classification maps often exhibit some degree of spatial autocorrelation [1], [2], so if we are querying against only the relative distribution of classes rather than their precise location, do we need to process every pixel? In other words, to what extent does downsampling (upscaling) to save resources affect query accuracy?

The multispectral instrument (MSI) of the Copernicus Sentinel-2 mission captures data from 13 optical bands at spatial resolutions of 10 m–60 m (ground sampling distance).

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)

No data	Water
Saturated	Unknown
Dark	Cloud (medium chance)[a]
Shadow	Cloud (high chance)[a]
Vegetation	Cloud (cirrus)[a]
Bare soil/built-up	Snow

Fig. 1. SCL Classes. [a] Combined in our analysis. Source: [4].

From these, Level-2A (L2A) analysis-ready products are created, typically with the Sen2Cor software [3]. One component of such products is the Scene Classification Layer (SCL)—an auxiliary information layer generated with a rules-based classifier, comprising twelve classes covering various land cover types, cloud, and anomalous readings (see Fig. 1) [4].

Cloud Optimized GeoTIFF (COG) is a geospatial raster file format where data are stored as tiles, such that spatial subsets can be read without needing to load the entire file. This subsetting works over a network using, e.g., HTTP range requests. The underlying GeoTIFF offers pyramid/overview features that allow access to data at different scales. This can drastically lower the data transfer and processing costs of geospatial analysis [5]. But whether this can be leveraged depends on the use case, two factors of which are the size of the user's area of interest (AoI) and the scale (areal unit size).

The Microsoft Planetary Computer (MSPC) platform [6] hosts a Sentinel-2 L2A data archive, including the SCLs. The asset data is stored as COGs, which can be located via a Spatiotemporal Asset Catalog (STAC) metadata search and retrieved via HTTP(S). The SCL data is stored as a raster with 20 m spatial resolution [7]. A typical S2 image granule spans approximately 110 km × 110 km [8]. An entire SCL raster, therefore, has around 30 megapixels. The class distribution for the entire scene is actually pre-computed and stored in the metadata, accessible by parameters such as `s2:vegetation_percentage` and `s2:snow_ice_percentage` [9]. This is useful if a user's AoI matches the granule's footprint, but this is often not the case, e.g., for analyses based on administrative boundaries.

The role of scale in remote sensing has been covered extensively in [10], but this was before the time of the Sentinel

This article has been accepted for publication in IEEE Geoscience and Remote Sensing Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LGRS.2026.3720762

missions, COGs, and Big Earth Data. Later works have revisited the topic, exploring different resampling approaches and demonstrating the effects of downsampling in a range of other settings, such as classification and land cover accuracy on a pixel-by-pixel basis [11]–[14]. While these previous works characterize the effects of downsampling across various scenarios, the goal of our work was to determine the limits for a given error threshold, allowing a user to semantically search petabytes of data efficiently and accurately.

A preliminary version of this work was presented at MIGARS 2025 [15]. This article further explores the ideas presented with an additional study area, new experimental results, and further technical analysis.

II. Methods

A. Study areas and period

At first [15], two study areas were selected: an area around Ribera del Duero, Spain (between 41.47°N to 42.57°N latitude and 3.57°W to 5.40°W longitude), and Stockerau, Lower Austria (between 47.78°N to 48.87°N latitude and 15.01°E to 16.65°E longitude). These represent two agricultural areas in Europe with differing climate and land use/land cover characteristics. Both contain many small, spatially heterogeneous agricultural parcels. The study period was 1st June to 1st July 2023, when the expected conditions for these areas of Europe would be a mix of cloudy and clear days, with significant vegetation growth. Since the initial work, an additional study area with different characteristics was selected—northern Palawan island, Philippines, and its surroundings (between 10.78°N to 11.86°N latitude and 118.86°E to 119.96°E longitude). This area covers islands of various sizes with varying amounts of tropically vegetated mountainous terrain, and sandy beaches. Despite the study period falling in the rainy season, there were several partially clear images.

B. Downsampling analysis

Based on the study areas, series of synthetic AoIs were generated: square tiles of side lengths (sizes) varying from 1 to 60 km, using the coordinate reference system (CRS) of the corresponding Universal Transverse Mercator (UTM) grid zone. For reference purposes, these were grouped by size. An overview is provided in Table I.

TABLE I
SUMMARY OF AOIS AND VALID IMAGE SAMPLES

AoI tile size (m)	Size group	# AoIs	Valid samples[b]
1000	Small	165	1859
2000	Small	49	538
3000	Small	25	270
4000	Small	16	168
6000	Medium	51	429
8000	Medium, Large	63	533
10 000	Large	15	128
12 000	Medium, Large	31	259
20 000	Large, Extra-large	112	781
40 000	Extra-large	11	61
60 000	Extra-large	2	10

[b] Matching, complete image chips.

For each AoI, a STAC search was performed. For each matching image, the SCL was windowed to the AoI, and the class distribution (as percentages) at native resolution (20 m) was calculated. The SCL was then queried at a series of lower resolutions using regular nearest-neighbor downsampling. The resolutions were selected using a loosely exponential pattern, representing a variety of analytical scales: 50 m, 100 m, 200 m, 500 m, and 1000 m.

The class distributions were calculated for each of the downsampled variants and compared against that of the native resolution. The maximum absolute difference in percentages, which we refer to as the maximum classification error (MCE), is our key metric; it directly informs the accuracy of our use-case queries. An illustrative, hypothetical example of the MCE calculation, and its application in a real image, are given in Table II and Fig. 2, respectively.

TABLE II
HYPOTHETICAL CLASS DISTRIBUTION CHANGE

	Vegetation	Bare soil	Cloud
Native (20m)	10%	10%	80%
Downsampled (100m)	8%	11%	81%
Difference	−2%	+1%	+1%
Absolute difference	2%[c]	1%	1%

[c] Maximum classification error

Some preliminary results showed low spatial autocorrelation among the cloud classes (8, 9, 10), producing large classification errors after downsampling. However, our use cases suggested that an aggressive approach to cloud filtering would be appropriate, and these classes were combined via summation of their percentages.

Sometimes an AoI would only partially overlap with the footprint of an image granule, yielding areas of 'no data', or the image data itself would contain 'no data' values. These were excluded from the class distribution comparisons. Valid samples, as in Table I, therefore refer to image chips that contain zero 'no data' values.

The maximum classification errors were compared across all AoIs for each tile size and downsampling resolution. From this, we could determine the distributions of classification errors due to downsampling, and where the maximum exceeded a user-defined threshold/tolerance (1% as an initial suggestion).

A demonstration of the effect of spatial distribution—naturally occurring versus random—on classification error due to downsampling was also performed. This is detailed in section S1 of the supplementary materials.

The analysis was performed with Python 3.14 in a `marimo` notebook [16], using packages `pystac_client` [17] and `stackstac` [18] for data loading, packages `xarray` [19], `pandas` [20], `geopandas` [21] and `scipy` [22] for data processing and analysis, and `holoviews` [23] for visualization. This code was run on an HP EliteDesk workstation with an Intel Core i5-8500 processor (3 GHz, 6 cores) and a 256 GB

This article has been accepted for publication in IEEE Geoscience and Remote Sensing Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LGRS.2026.3720762

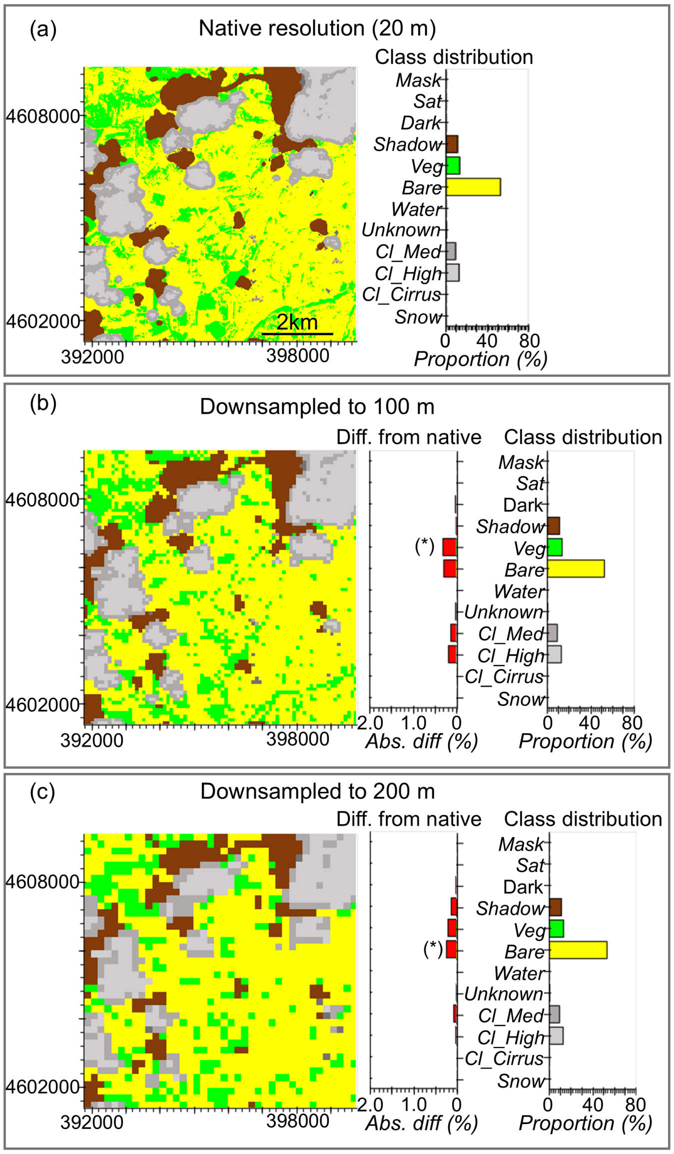

Fig. 2. SCL and the class distribution for an 8 km × 8 km extract of Sentinel-2 scene S2B_MSIL2A_20230605T110619_R137_T30TUM_20230605T163049, at native resolution (a), downsampled to 100 m × 100 m (b), and downsampled to 200 m × 200 m (c). The maximum classification error (MCE) is labelled (*). Coordinates are EPSG:32630. Contains modified Copernicus Sentinel data (2023).

solid-state system drive, in a Windows Subsystem for Linux (WSL) Docker container with 24 GB of memory available.

C. Performance

For each AoI, we timed the duration of querying the matching SCL assets from MSPC and calculating the class distributions at a given resolution. This consisted of remote cloud computation, storage access and network transfer, as well as local computation. The measurements are, therefore, highly subject to external conditions such as network, storage and computation contention, and various types of caching. Samples containing 'no data' values were included in these measurements. There were around 5–10 images matching each AoI in the date range; the duration to load and process all of these was recorded and divided by the number of images, giving a mean per-scene runtime. The number of pixels in each image was also recorded, as this, rather than physical size, determined the amount of data involved. These ranged from single-pixel images (as in a heavily downsampled small AoI) to 9-megapixel images (as in a 60 km size tile at native resolution). The Python standard library's `time` module was used. Note that the duration of the initial STAC search to find the matching images for the AoI was not included.

A simple linear regression model was constructed to estimate the query performance across the range of scene sizes. The per-scene runtime calculation amortizes system latency across all scenes in a query, which may be processed concurrently. In an attempt to compensate for the variability in latencies, a baseline level was determined from the 95th percentile runtime of scenes comprising 256 pixels or fewer (a computationally trivial amount compared to the larger scenes). Values falling below this threshold were excluded from the model.

III. Results

Fig. 3 shows the distributions of the maximum classification errors at a downsampling resolution of 100 m for all AoIs. Here it can be seen that maximum classification errors approach 20% at tile sizes of 1 km but fall under 1% at tile sizes of 10 km. For readability, similarly detailed presentations of the results for the remaining resolutions have been omitted. However, the key results are summarized in Table III, which shows the maximum downsampling resolution for each tile size before the maximum classification error exceeds a tolerance threshold of 1%. Note how the maximum downsampling resolution rises to 500 m at AoI tile sizes of 60 km.

A summary of the performance measurements by pixel count (grouped/binned exponentially due to the wide range of magnitudes) is given in Table IV, and the key characteristics are visualized in Fig. 4. Individual data are shown in Fig. 5. Together, these show that the average runtime steadily increases with the number of pixels, but with high variance (as indicated by the rolling interquartile range).

A baseline level of 0.0792 s per scene was calculated from samples of 256 or fewer pixels, above which a simple least-squares linear regression model fits the data ($N = 481$) with a slope of 5.27×10^{-8} s/pixel, y-intercept of 0.094 s, and an R^2 of 0.46. Despite the relatively weak R^2, the query speed-up from downsampling can be estimated. These are shown for each tile size in Table III. Downsampling offers a mild benefit to query runtimes for AoIs of 10 km or less, then reductions quickly increase with larger AoI sizes.

IV. Discussion and Conclusion

Our results show that downsampling can significantly reduce the data usage and runtime of aggregated, distribution-based semantic queries on land cover classes, if afforded a small tolerance for error: up to a 625× data reduction

This article has been accepted for publication in IEEE Geoscience and Remote Sensing Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LGRS.2026.3720762

TABLE III
MAXIMUM DOWNSAMPLING RESOLUTION BEFORE MAXIMUM CLASSIFICATION ERROR EXCEEDS 1% ("N/A", OTHERWISE), AND ASSOCIATED DATA AND ESTIMATED PER-SCENE QUERY RUNTIME REDUCTIONS

Tile size (m)	Max. downsampling resolution (m)	Pixels (native)	Pixels (downsampled)	Data reduction (multiple)	Est. runtime (native)	Est. runtime (downsampled)	Est. runtime reduction (multiple)
1000	N/A	2500	N/A	N/A	0.0951	N/A	N/A
2000	N/A	10 000	N/A	N/A	0.0955	N/A	N/A
3000	N/A	22 500	N/A	N/A	0.0962	N/A	N/A
4000	50	40 000	6400	6.25	0.0971	0.0953	1.02
6000	50	90 000	14 400	6.25	0.0997	0.0957	1.04
8000	50	160 000	25 600	6.25	0.1034	0.0963	1.07
10 000	100	250 000	10 000	25	0.1081	0.0955	1.13
12 000	100	360 000	14 400	25	0.1139	0.0957	1.19
20 000	200	1 000 000	10 000	100	0.1477	0.0955	1.55
40 000	200	4 000 000	40 000	100	0.3057	0.0971	3.15
60 000	500	9 000 000	14 400	625	0.5692	0.0957	5.95

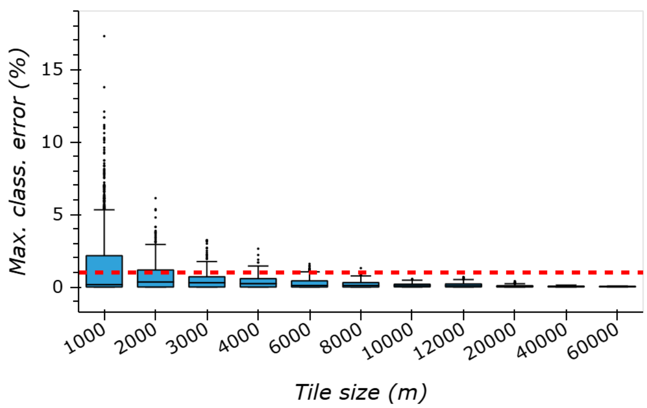

Fig. 3. Distributions of the maximum classification errors at a downsampling resolution of 100 m across all AoIs. Quartiles are represented by the boxes, data within 1.5 × interquartile range by whiskers, and outliers as dots. The dashed red line marks a 1% error tolerance threshold.

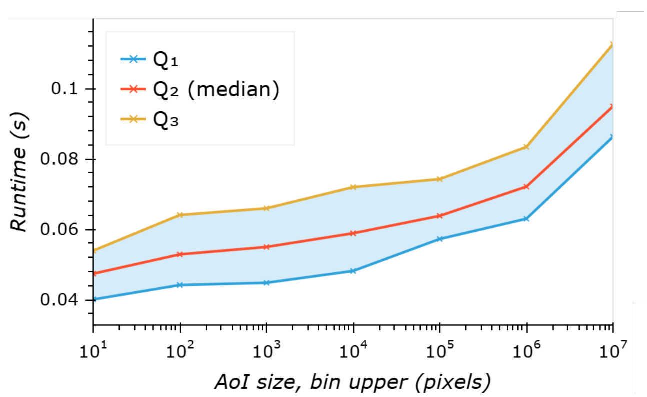

Fig. 4. Query runtime (mean per scene) quartiles (Q_{1-3}), binned by AoI pixel count.

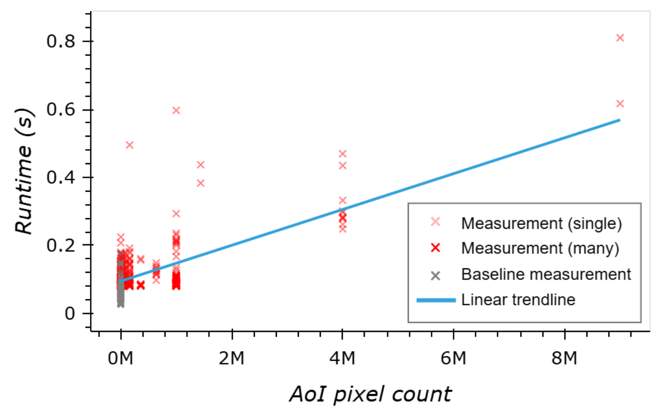

Fig. 5. Query runtime, mean per-scene. The linear trendline has a slope of 5.27×10^{-8} s/pixel, y-intercept of 0.094 s, and an R^2 of 0.46; $N = 481$. The baseline threshold is 0.0792 s, below which 1188 samples were excluded from the model.

TABLE IV
SUMMARY OF MEAN-PER-SCENE QUERY RUNTIMES

Pixel count	Queries	Min. (s)	Max. (s)	Q_1 (s)	Q_2 (s)	Q_3 (s)
1 to 10	404	0.026	0.180	0.040	0.047	0.054
10 to 100	385	0.026	0.173	0.044	0.053	0.064
10^2 to 10^3	863	0.026	0.173	0.045	0.055	0.066
10^3 to 10^4	630	0.026	0.225	0.048	0.059	0.072
10^4 to 10^5	586	0.026	0.178	0.057	0.064	0.074
10^5 to 10^6	245	0.050	0.495	0.063	0.072	0.083
10^6 to 10^7	127	0.074	0.811	0.086	0.095	0.113

Q_1, Q_2, and Q_3 are the quartiles, lowest to highest.

and estimated 6× decrease in query runtime for the largest AoIs in our experiment, with 1% error tolerance. For small-to-medium-sized AoIs, the benefits were small. In terms of performance, the smaller number of pixels meant they occupied the 'sill' where runtime was predominantly made up of network latencies rather than data processing. As they made up most of the observations, this also led to the moderately weak R^2 of 0.46 in the linear prediction model. The high range of the baseline measurements (from 0.026 to 0.18 s per scene) lends a high degree of uncertainty to measurements across the range of AoI pixel counts. This shows the model should be treated as indicative only, and should not be used for extrapolating performance improvements beyond this range, or making precise statements on performance. Our initial experiment showed a query runtime reduction of up to 12× [15]; a previous version of this work (not published) showed

This article has been accepted for publication in IEEE Geoscience and Remote Sensing Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LGRS.2026.3720762

a 15× reduction; here we obtain a 6× reduction. These differences indicate high sensitivity to external conditions, and the performance benefits at lower pixel counts appear only in aggregate across a large number of queries.

In terms of class distribution error, there are likely more nearly optimal resolutions to which AoIs can be downsampled, especially for the smallest AoIs (which are more sensitive to downsampling) and those larger than we tested. The experiment was kept constrained to avoid excessive use of cloud resources (a full run completes in approximately 45 minutes).

Only uniform, regular grid downsampling was used here. Other methods, such as the adaptive approach of Mîrț et al. [14], could perform better in terms of maintaining class distribution. Applied at query time, this would lose the data handling performance benefits tied to existing COG files' pyramids; however, if it were chosen as the pyramid resampling algorithm at COG creation time, future queries would likely benefit.

If the COG internal tile size and overview structure are known ahead of time, adjusting the downsampling resolution to match these would avoid an additional resampling step, leading to additional performance gains. As a corollary, our results could inform the tile sizing when generating future COGs.

Given the small performance gains at small-to-medium-sized AoIs, the accuracy tradeoff means downsampling using the thresholds given in Table III may not be suitable for some use cases. However, the reduced memory usage, which is independent of external conditions, should also be taken into consideration.

Running the experiment on the additional Philippines study area provided additional data points, but did not impact the final downsampling thresholds. Nevertheless, repeating the approach on further, geographically diverse study areas would be a useful extension of this work. As would applying it to land cover maps beyond the Sentinel-2 SCL, or other binned/categorical data types.

Data and Software Availability

The Sentinel-2 source imagery data reside in the Microsoft Planetary Computer system [7]. However, the source code and key intermediate files for this analysis are available, under an MIT license, in supplementary file `eaggq-source-1.1.1.zip`, and online at https://github.com/lukesdm/eaggq-materials/tree/v1.1.1 (see file `README.md` for instructions).

Generative AI Declaration

In this work, generative AI was used for assistance with formatting, grammar, and coding. Code was rarely directly integrated *as is*, and is labeled as such, where it was. All was manually verified.

Acknowledgment

Many thanks go to our colleague, Felix Kröber, whose earlier explorations and Jupyter notebooks on Sentinel-2 STACs and the SCL were a great help; to the Microsoft Planetary Computer team for making their service available to us; and to our colleague Leizel De La Cruz for insights on the Philippines study area.

References

[1] W. R. Tobler, "A Computer Movie Simulating Urban Growth in the Detroit Region," *Economic Geography*, Jun. 1970, publisher: Routledge.
[2] P. J. Curran, "The semivariogram in remote sensing: An introduction," *Remote Sensing of Environment*, vol. 24, no. 3, pp. 493–507, Apr. 1988.
[3] M. Main-Knorn, B. Pflug, J. Louis, V. Debaecker, U. Müller-Wilm, and F. Gascon, "Sen2Cor for Sentinel-2," in *Image and Signal Processing for Remote Sensing XXIII*, L. Bruzzone, Ed., vol. 10427, International Society for Optics and Photonics. SPIE, 2017, p. 1042704. [Online]. Available: https://doi.org/10.1117/12.2278218
[4] European Space Agency, "Sentinel-2 Level-2A Algorithm Theoretical Basis Document," ESA – European Space Agency, Frascati, Italy, Technical Report S2-PDGS-MPC-ATBD-L2A, Nov. 2021, version: 2.10.
[5] Cloud Optimized GeoTIFF Community, "Cloud Optimized GeoTIFF," 2025. [Online]. Available: https://cogeo.org/
[6] Microsoft Open Source, M. McFarland, R. Emanuele, D. Morris, and T. Augspurger, "microsoft/PlanetaryComputer: October 2022," Oct. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7261897
[7] Microsoft, "Sentinel-2 Level-2A | Microsoft Planetary Computer," accessed: May 13 2025. [Online]. Available: https://planetarycomputer.microsoft.com/dataset/sentinel-2-l2a
[8] European Space Agency, "Sentinel-2 Products Specification Document," ESA – European Space Agency, Frascati, Italy, Tech. Rep. S2-PDGS-TAS-DI-PSD, Mar. 2021.
[9] M. Mohr and P. Varner, "stac-extensions/sentinel-2," Nov. 2023, version: v1.0.0. [Online]. Available: https://github.com/stac-extensions/sentinel-2
[10] S. Weigel, "Scale, Resolution and Resampling: Representation and Analysis of Remotely Sensed Landscapes Across Scale in Geographic Information Systems." *LSU Historical Dissertations and Theses*, Jan. 1996.
[11] H. Wu and Z.-L. Li, "Scale Issues in Remote Sensing: A Review on Analysis, Processing and Modeling," *Sensors*, vol. 9, no. 3, pp. 1768–1793, Mar. 2009. [Online]. Available: https://www.mdpi.com/1424-8220/9/3/1768
[12] R. Raj, N. A. S. Hamm, and Y. Kant, "Analysing the effect of different aggregation approaches on remotely sensed data," *International Journal of Remote Sensing*, Jul. 2013.
[13] K. Xu, Q. Tian, Y. Yang, J. Yue, and S. Tang, "How up-scaling of remote-sensing images affects land-cover classification by comparison with multiscale satellite images," *International Journal of Remote Sensing*, Apr. 2019.
[14] A. Mîrț, J. Reiche, J. Verbesselt, and M. Herold, "A Downsampling Method Addressing the Modifiable Areal Unit Problem in Remote Sensing," *Remote Sensing*, vol. 14, no. 21, p. 5538, Jan. 2022.
[15] L. McQuade, M. Sudmanns, and D. Tiede, "Efficient aggregate land cover queries with cloud-optimized raster formats," in *2025 International Conference on Machine Intelligence for GeoAnalytics and Remote Sensing (MIGARS)*, 2025, pp. 1–4.
[16] A. Agrawal and M. Scolnick, "marimo," Feb. 2026, version: v0.20.2. [Online]. Available: https://github.com/marimo-team/marimo
[17] M. Hanson et al., "stac-utils/pystac-client," Jul. 2025, version: v0.9.0. [Online]. Available: https://github.com/stac-utils/pystac-client
[18] G. Joseph, "gjoseph92/stackstac," Aug. 2024, version: v0.5.1. [Online]. Available: https://github.com/gjoseph92/stackstac
[19] S. Hoyer et al., "xarray," Nov. 2025, version: v2025.11.0. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.15110615
[20] The pandas development team, "pandas-dev/pandas: Pandas," Sep. 2025. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.17229934
[21] J. V. den Bossche et al., "geopandas/geopandas: v1.1.2," Dec. 2025. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.18024156
[22] R. Gommers et al., "scipy/scipy: v1.17.0," Jan. 2026. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.18209846
[23] P. Rudiger et al., "holoviz/holoviews: Version 1.22.1," Dec. 2025. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.17831576