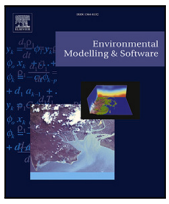
Environmental Modelling and Software 183 (2025) 106257

Contents lists available at ScienceDirect

Environmental Modelling and Software

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



Position Paper

Enabling coastal analytics at planetary scale ☆

Floris Reinier Calkoen [a,b,*], Arjen Pieter Luijendijk [a,b], Kilian Vos [c], Etiënne Kras [a], Fedor Baart [a,b]

[a] Deltares, Boussinesqweg 1, Delft, 2629 HV, Zuid-Holland, The Netherlands
[b] Delft University of Technology, Stevinweg 1, Delft, 2628 CN, Zuid-Holland, The Netherlands
[c] Water Research Laboratory, 110 King Street, Manly Vale, 2093, New South Wales, Australia

ARTICLE INFO

Dataset link: https://github.com/TUDelft-CITG/coastpy, https://zenodo.org/records/14056925

Keywords:
Coastal analytics
Cloud technology
Coastal change
Coastal monitoring
Satellite-derived shorelines
Low elevation coastal zone
Data management

ABSTRACT

Coastal science has entered a new era of data-driven research, facilitated by satellite data and cloud computing. Despite its potential, the coastal community has yet to fully capitalize on these advancements due to a lack of tailored data, tools, and models. This paper demonstrates how cloud technology can advance coastal analytics at scale. We introduce GCTS, a novel foundational dataset comprising over 11 million coastal transects at 100-m resolution. Our experiments highlight the importance of cloud-optimized data formats, geospatial sorting, and metadata-driven data retrieval. By leveraging cloud technology, we achieve up to 700 times faster performance for tasks like coastal waterline mapping. A case study reveals that 33% of the world's first kilometer of coast is below 5 m, with the entire analysis completed in a few hours. Our findings make a compelling case for the coastal community to start producing data, tools, and models suitable for scalable coastal analytics.

1. Introduction

Coastal science has entered a new era of data-driven research (Vitousek et al., 2023a), facilitated by the opening up of historical satellite data catalogs (Wulder et al., 2022) and advances in data processing (Dean and Ghemawat, 2008), which have been integrated into cloud-based geospatial data analysis platforms (e.g., Gorelick et al., 2017). In coastal science, the potential of Earth-observing satellite data was showed by global analyses to coastal change (e.g., Luijendijk et al., 2018; Murray et al., 2018) and since then the coast is studied at increasing detail using satellite-derived data products (e.g., Warrick et al., 2023).

Currently we can distinguish between two distinct analysis strategies: one aims for global coverage, typically compromising accuracy for spatial extent ("everywhere"), while the other prioritizes accuracy ("anywhere"). Although not all coastal analyses have to be run at broad spatial scales (e.g., Mikkelsen et al., 2024), there are several reasons for why we need coastal analyses at extensive spatial scales. Particularly since the advent of Geospatial Information System (GIS) in coastal science, it has been acknowledged that analyses at scale facilitate integrated and systematic approaches to coastal classification (Finkl, 2004). It also supports development of diverse coastal management strategies at varying spatial scales (Cooper and McLaughlin, 1998). Moreover, while details and planning of coastal management often happen at local or regional levels, they are supported by legislation at national or international levels (Wong et al., 2014). Finally, analyses at scale facilitate intercomparisons between different regions, enabling peer-to-peer learning, where local coastal management practices that have been adopted successfully can be shared.

The two distinct analysis strategies ("everywhere" vs "anywhere") can be illustrated by contrasting some coastal monitoring tools in more detail. On one hand, analyses at scale ("everywhere") typically use a cloud platform to process petabyte-scale satellite data catalogs by condensing stacks of individual imagery into composites (e.g., Luijendijk et al., 2018; Mao et al., 2021) or cloud-free mosaics (e.g., Hulskamp et al., 2023). These approaches rely on methods that are available on the cloud platform, so that they can be incorporated in server-side compute. While such strategies manage large volumes of data by efficiently processing data in close proximity of where it is stored, it inherently limits the temporal depth and/or restricts the analysis methods to those available on the cloud platform. On the other hand, approaches that process each image in the historical catalog using more sophisticated processing routines or algorithms (e.g., Buscombe and

☆ This research was funded by European Commission SOCIETAL CHALLENGES - Climate action, Environment, Resource Efficiency and Raw Materials as part of CoCliCo (Grant agreement ID: 101003598).

* Corresponding author.
E-mail addresses: floris.calkoen@deltares.nl (F.R. Calkoen), arjen.luijendijk@deltares.nl (A.P. Luijendijk), voskilian@gmail.com (K. Vos), etienne.kras@deltares.nl (E. Kras), fedor.baart@deltares.nl (F. Baart).

https://doi.org/10.1016/j.envsoft.2024.106257
Received 7 June 2024; Received in revised form 22 October 2024; Accepted 25 October 2024
Available online 8 November 2024

Ritchie, 2018; Vos et al., 2019; Al Najar et al., 2023; Muir et al., 2024), have traditionally been confined to local studies or demanded substantial effort (e.g., Vos et al., 2023a; Vitousek et al., 2023b; Castelle et al., 2024). Such approaches typically involve downloading all data from the server to a machine with all required software to run the analysis installed. In summary, analyses focusing on larger geographic areas often sacrifice either temporal depth, algorithmic flexibility, or both, whereas those prioritizing more detailed analyses would face problems to scale up to larger areas if that is desired.

Historically, the importance of data-proximate, scalable compute on rasters (Baumann, 1993) and the ability to query data onto a uniform grid for comparison (Cornillon et al., 2003) have been well acknowledged. These principles are now typically integrated into geospatial data processing platforms (e.g., Medvedev et al., 2016; Raoult et al., 2017). Later, Gorelick et al. (2017) have arguably revolutionized Earth science by building a geospatial data analytics platform on the public cloud, where users can write and execute code to efficiently perform more complex analyses on vast datasets by co-locating data and compute infrastructure as well as providing a high-level scripting environment. Despite its success, Google Earth Engine (GEE) remains a platform with limited flexibility and modularity (Abernathey et al., 2021) and nowadays there are also more open, flexible, community-driven efforts that aim to facilitate big environmental data analysis. Both Open Data Cube (ODC) (Killough, 2018) and Pangeo (Abernathey et al., 2017) emphasize open-source development and collaboration, with active communities across all continents. ODC focuses on providing a structured framework for managing and analyzing satellite data, while the aim of Pangeo is broader, with its community contributing to a coherent software ecosystem that aims to enable big data geoscience. ODC has particularly been successful in Australia, where Digital Earth Australia (DEA) (Gavin et al., 2018) have also enabled applications in coastal monitoring at scale (Bishop-Taylor et al., 2021). Although ODC is open, flexible and capable of handling large computations, it has primarily been implemented on high-performance clusters at a national level (Killough, 2018). Both ODC and Pangeo leverage similar open-source software, but Pangeo, with its larger ecosystem, also supports cloud-based platforms,[1] where one can conduct highly specialized analytics. These platforms are built on open, scalable software and can integrate diverse data sources like satellite imagery, climate data and other geospatial features, marking a shift towards more open, flexible and scalable geospatial data analytics.

We contend that if the coastal community wants to gain new insights into urgent coastal challenges at extensive scale, without compromising accuracy or spatio-temporal resolution, it will probably have to start producing tools, models, and data that are suitable for scalable analytics. In this paper, we share our experience in using cloud technology to advance coastal science at scale, effectively bridging the gap from "everywhere" towards "anywhere".

This paper is divided into three parts. The first part discusses data-proximate computing through the implementation of coastal waterline mapping in the cloud. It highlights the advantages of data-proximate computing and explores how a flexible software stack (Pangeo) facilitates scalable coastal analytics. The second part compares different data storage strategies for our novel GCTS, a foundational dataset, consisting of cross-shore coastal transect system at 100-m alongshore resolution, that is now made publicly available. The experiments demonstrate how cloud-optimized data formats are significantly more efficient than traditional file formats, making them critical for coastal analytics at scale. In this part we also highlight the importance of standardized metadata specifications. Finally, the third part demonstrates how flexible, scalable compute (part 1), combined with cloud-optimized data exposed through standardized metadata specifications (part 2), enable high-resolution coastal analytics (100-m alongshore) at planetary scale.

As a use case, we compute the percentage of land within the first kilometer of the coastal zone that is below 5 m above mean sea level. With this work, we aim to show that coastal analytics can be performed at global scale, without compromising accuracy or spatio-temporal resolution, in very reasonable compute times. Can we make a compelling argument for the coastal community to start producing tools, models, and data that are suitable for scalable coastal analytics?

2. Methodology

In this section, we present our framework for conducting high-resolution, planetary-scale coastal analytics. We focus on two experiments: data-proximate coastal waterline mapping and strategies for cloud-native data release. Insights gained from these experiments inform a subsequent case study, which extracts elevation data over more than 11 million coastal transects, illustrating how scalable tools, models, and data can advance coastal science. Overall, insights from experiment 1 and experiment 2 enable the case study on coastal elevation mapping. In this section, the methods are described by topic, with some methods used in multiple experiments as well as in the case study. Fig. 1 provides an overview of the described architecture, encompassing all methods detailed in this section.

2.1. Global coastal transect system

Cross-shore coastal transects are essential to coastal monitoring, offering a consistent reference line to measure coastal change, while providing a robust foundation to map coastal characteristics and derive coastal statistics thereof. In this work, we introduce the GCTS, a novel foundational dataset compromising more than 11 million cross-shore coastal transects uniformly spaced at 100-m intervals along the shore. In comparison to previous efforts (Luijendijk et al., 2018; Bishop-Taylor et al., 2021; Vos et al., 2023a), this system has several advantages. The dataset has global coverage at 100-m alongshore resolution, for all OpenStreetMap (OSM) coastlines (80° S - 84° N) that are longer than 5 kilometers. We decided to define transects at 100-m alongshore resolution because this has shown to be effective for studying coastal dynamics at broad spatial scales (Bishop-Taylor et al., 2021; Vos et al., 2023a); it aligns well with the typical resolution of public satellite imagery (~10–30 m); and, this resolution is also used in numerical modeling studies (Roelvink et al., 2020). The transect system is derived from a recent (2023-01-23) generalized OSM coastline,[2] that was specifically prepared,[3] with an optimal balance between smoothing and simplification (Hormann, 2014) for a coastal cross-shore transect system at this 100-m alongshore resolution. Expert evaluations and visual comparisons with existing systems, such as the manually digitized CoastSat transects for the Pacific Basin (Vos et al., 2023a), confirm that zoom level 9 of the generalized OSM coastline provides the most accurate transects at this scale. Furthermore, using the most recent coastline data allows us to incorporate the latest crowd-sourced data from the OSM project. Finally, the transects are derived in their (Universal Transverse Mercator (UTM)) projection, a conformal projection that preserves angles locally, maintaining a uniform length of 2000 m and a spacing of 100 m apart alongshore, effectively correcting the zonal (latitude) distortions present in earlier global transect systems. GCTS is licensed under CC BY 4.0 licence, which means that you are free to share and adapt the dataset, as long as you give appropriate credit (i.e. cite this paper).

In this first release, we also add administrative boundaries[4] and a north bearing—the angle measured in degrees in a clockwise direction

[1] Notable examples include Microsoft Planetary Computer (MSPC), Coiled and Earthmover.

[2] OSM data is available under the Open Database License (ODbL) at https://openstreetmap.org.

[3] The generalized coastline was produced by Imagico, DE.

[4] Administrative boundaries are extracted from Overture Maps.

from the north pole. The data is released following best practices, as derived in experiment 2 (See Section 3.2) to facilitate convenient, efficient data retrieval, while ensuring that each chunk comfortably fits in the memory of a regular personal computer. The transects are stored in cloud-optimized GeoParquet format within a public cloud container[5] and are available as part of the Coastal Climate Core services (CoCliCo) Spatio-Temporal Asset Catalog (STAC) catalog[6] under the collection ID "gcts". For those less familiar with cloud services, the data is also available on Zenodo, at https://zenodo.org/records/14056925.

2.2. Software stack

The methods we use to conduct coastal analytics at extensive spatial scales are deeply integrated with core-packages of the Pangeo stack for big environmental data analysis. The data processing routines employed typically utilize the data models Pandas (McKinney, 2010) and Xarray (Hoyer and Hamman, 2017), which, when integrated with geospatial extensions like GeoPandas and Rioxarray, enable advanced geospatial analysis of both raster and vector data, respectively. In coastal analytics, many of the data processing steps, such as computing spectral indices, can be run in parallel. To leverage the efficiency of parallel processing, we utilize Dask, an open-source library for parallel and distributed computing (Rocklin, 2015). Dask distributes computational tasks across multiple workers, which can also operate on different nodes. This distribution is centrally managed using computational graphs, which provide a structured representation of the tasks. Unlike traditional for loops, which execute sequentially, Dask's graph system allows tasks to be executed concurrently, optimizing performance and reducing computation time. We also use DuckDB, an embeddable analytical relational database management system (Raasveldt and Mühleisen, 2019) to efficiently filter and retrieve tabular geospatial data from cloud object storage. To enhance the efficiency of distributed computing in coastal monitoring, we have refactored some routines from earlier coastal monitoring efforts (Luijendijk et al., 2018; Vos et al., 2019; Bishop-Taylor et al., 2021) into vectorized functions — operations applied simultaneously across entire arrays of data. This enables integration into the Dask data processing chain for more efficient parallel processing. Furthermore, by executing computations on a server nearby data storage and leveraging on-demand cloud compute infrastructure with tools like Dask Gateway or Dask Jobqueue, we can further reduce processing times by efficiently distributing tasks across a larger network of nodes. For example, when using a cloud compute infrastructure like MSPC, we are able to distribute tasks across more than 100 'workers' who collectively have access to approximately 800 GB of memory. In such distributed network, secondary data movement, such as coastal transects in the third experiment (See Section 3.3) is minimized by strategically scattering it over the network, allowing it to be referenced through pointers (addresses to the data location) rather than sending the actual data.

2.3. Data retrieval

Data is searched for per area of interest, data range or other attributes using STAC, a geospatial data specification that enables efficient localization of spatio-temporal data collections through its standardized Application Programming Interface (API). Data is retrieved from various STAC catalogs, including the MSPC and CoCliCo catalogs. The STAC collections are browsed, and relevant storage locations are parsed by the respective data models (See Section 2.2). For vector data, such as coastal transects, spatial joins or predicate pushdowns—a technique that filters data at the database level to improve query efficiency—on the bounding box attribute are used to optimize data retrieval. Raster data, such as satellite imagery, are lazily loaded, meaning that data is only loaded into memory when it is actually needed for computation. The data are read into Xarray using ODC-STAC, a Python library that is part of DEA software ecosystem. To minimize redundant Earth Observation (EO) data processing, imagery transmitted to multiple downlink stations or covering overlapping areas (Bauer-Marschallinger and Falkner, 2023) on the same orbital ground track is grouped by solar day, selecting only the first occurrence of each day. Precise pixel alignment is achieved by centering pixels on the coordinate grid, ensuring all coordinate axes are anchored at pixel centers.

2.4. Data processing

To avoid memory issues and create a scalable, dynamic, parallel data processing strategy, larger areas are divided into a hierarchical grid using so-called quadtiles,[7] a geo-data storage and indexing strategy. This strategy operates at specific zoom levels. For example, for processing Sentinel-2 (S2) tiles, we subdivide them into quadtiles at zoom level 10, which corresponds to an area of approximately 0.35 degrees longitude around the equator. Data processing is performed on a per-quadtile basis, with computational graphs constructed and executed in a single compute call per quadtile. This approach prevents redundant recalculations of intermediate results and ensures a scalable data processing workflow. The chosen zoom level for the quadtiles (zoom level 10) ensures that arrays fit within memory limits, which is essential for complex processing workflows like coastal waterline mapping (See Section 2.8.1). We typically also include coastline buffers into the processing pipeline to limit processing solely within our designated coastal region of interest. These buffers, are derived from the OSM coastline as of 2023-01-23, with a radius of 10 km, that is derived per UTM region. While processing, secondary data, such as these coastline buffers, are broadcast across the cluster so that this data is referenced through pointers (addresses to the data location) rather than transferring the actual data, further optimizing the processing efficiency.

2.5. Data partitioning

When storing data it is partitioned into manageable chunks, ranging from 100 to 200 MB for raster data and 500 MB for vector data, to enhance data interoperability for downstream coastal analytics. We spatially partition static coastal data using a strategy that compensates for coastline complexity, maintaining uniform partition sizes despite geographical variances in coastal geomorphology. The partitioning strategy begins by estimating the memory usage per geometry and its attributes. It then adds a quadkey (also known as a geohash), a geospatial index that encodes the location of an attribute in a standardized string, suitable for hierarchical binning at a given zoom level (e.g., level 12). Next, the data is sorted by quadkey. Finally, the data is recursively partitioned into chunks that do not exceed a set memory usage threshold. This approach ensures that local data density stays below a predefined partition size threshold. This method is exemplified in Appendix B, where the GCTS is partitioned, for demonstration purposes, into chunks that do not exceed 100MB. It is evident that areas with higher data density, such as Chile, result in denser partitions. Finally, we typically assign a bounding box attribute with the minx, miny, maxx, and maxy coordinates of the geometry. This attribute is added as a structured datatype to enable query optimization techniques, such as predicate pushdowns, which filter data at the database level to improve query efficiency.

2.6. Data release

All data are stored in cloud-optimized formats, to ensure efficient retrieval (Durbin et al., 2020) from cloud object storage services that

[5] az://coclico.blob.core.windows.net/gcts/release/<date>/*.parquet
[6] https://coclico.blob.core.windows.net/stac/v1/catalog.json
[7] https://wiki.openstreetmap.org/wiki/QuadTiles

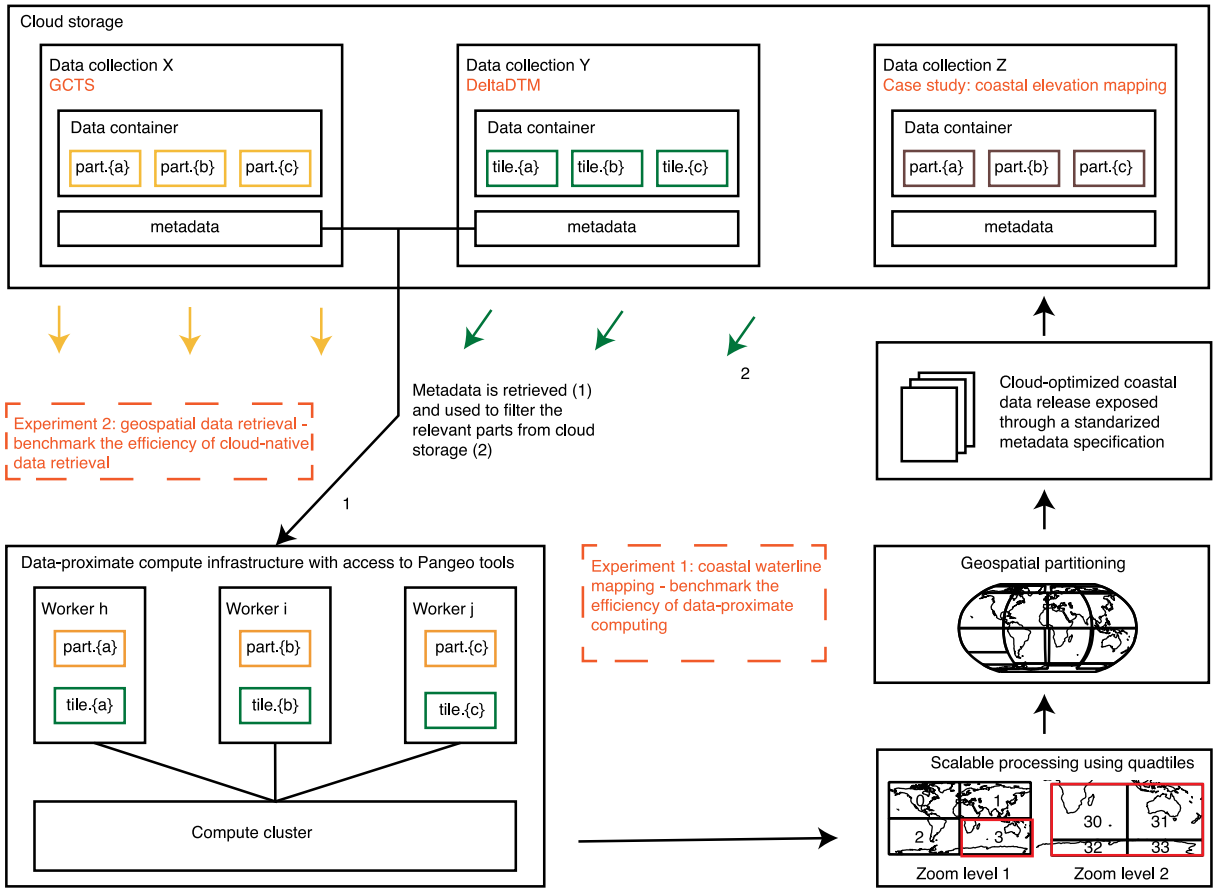


Fig. 1. The workflow architecture for coastal analytics at scale. Data are stored in cloud object storage and exposed through standardized metadata specifications for efficient retrieval. Compute infrastructure is set up in close-proximity of the data, facilitating scalable data processing via hierarchical quadtiles. The results are stored in geospatial partitions in the cloud and also described following a standardized metadata specification to enable the next iteration of downstream coastal analytics. This study includes two primary experiments: geospatial data retrieval and data-proximate coastal waterline mapping. Insights from these experiments inform a case study demonstrating coastal analytics at a planetary scale by combining coastal transects (GCTS) with elevation data (DeltaDTM).

can serve large volumes of data. Geospatial vector data, such as coastal waterlines and cross-shore transects, are saved as compressed Apache GeoParquet files, a widely-used columnar storage format. Time series data, obtained per station, such as SDS series, are also stored in the GeoParquet format, with additional attributes that enable efficient geospatial data retrieval (See Section 2.3). Raster data, such as binary water-occurrence maps without a temporal component, are stored as Cloud-Optimized GeoTIFF (COG), whereas n-dimensional array's (such as climate data cubes or raster data with temporal component) are stored in Zarr. The data sets are described in STAC collections, that are added to the CoCliCo STAC catalog.[8] Here, each data partition or chunk is referenced as a STAC item. The catalog and its collections are maintained with open-source glsstac-utils[9] software, which utilizes specific extensions like 'proj', 'eo', 'datacube' and 'stac-table' to validate metadata against community standards. To further optimize accessibility, a GeoParquet snapshot of all STAC items is included as a collection asset, eliminating the need for users to index each JSON file individually from the collection.

2.7. Usage

All methods and workflows are bundled in CoastPy, a domain-specific Python package that contains tools for scalable, cloud-based coastal analytics, that, especially when combined with cloud computing, can efficiently process large amounts of data. CoastPy is available on the PyPI repository and is actively maintained at https://github.com/TUDelft-CITG/coastpy. It supports Python ≥ 3.11 and is compatible with macOS, Windows, and Linux. Detailed installation instructions and usage guidelines are provided in the README file and documentation. The repository includes tutorial notebooks that introduce straightforward applications of coastal analytics at scale. Released under the MIT License, CoastPy promotes open-source collaboration and development. We plan to expand the repository with tools for coastal Machine Learning (ML) and other advancements, and we welcome contributions from the community to this cloud-based coastal analytics initiative.

2.8. Experiments

To study the potential of cloud-native workflows for coastal analytics, we conducted several experiments focusing on data-proximate computing and data storage strategies. Experiment 1 is about coastal waterline mapping near the physical storage of EO data and experiment 2 benchmarks various cloud-optimized data access strategies using a standard geospatial data retrieval pattern. Additionally, in a subsequent case study, we demonstrate the potential of cloud-native workflows for coastal analytics at planetary scale by studying the distribution of elevation in the coastal zone. All experiments were performed using a standard Pangeo container, that a was run on MSPC, which provided access to a Dask Gateway cluster with adaptive scaling. The cluster was configured to provide 8 GB of RAM per Dask worker, with the

[8] See https://coclico.blob.core.windows.net/stac/v1/catalog.json.
[9] See https://github.com/stac-utils.

number of workers dynamically adjusting between 2 and 100 based on processing workload and available compute. Dask acts as a higher-level tool that efficiently manages and distributes Python workflows across available compute resources. We emphasize that this a modular approach, where compute is separated from storage, that is not pertained to MSPC, but can be orchestrated at any cloud provider, (SLURM-based) High Performance Computing (HPC) cluster or even an ordinary personal computer.

2.8.1. Experiment 1: Coastal waterline mapping

The purpose of this experiment on coastal waterline mapping is to assess the efficiency of data-proximate cloud-computing for coastal monitoring by orchestrating coastal waterline mapping routines next to where the satellite data is stored. Given that shoreline monitoring has de-facto become standard practice in coastal science (Vos et al., 2023b), we expect that this experiment serves as a relatable example for most coastal practitioners.

In this experiment, the S2 archive of publicly available imagery (level-1 A surface reflectance) is efficiently retrieved (See Section 2.3) from the MSPC STAC catalog. To minimize redundant EO data processing, imagery transmitted to multiple downlink stations or covering overlapping areas (Bauer-Marschallinger and Falkner, 2023) on the same orbital ground track is grouped by solar day, selecting only the first occurrence of each day. The blue, green, red, NIR, SWIR16, and SCL classification bands from the S2 catalog are retrieved, while precise pixel alignment is achieved by centering pixels on the coordinate grid. Following Vos et al. (2023b), the SWIR16 band is aligned with other spectral bands using bilinear resampling, since this strategy better captures linear features such as waterlines. The S2 SCL layer, a 10-class land cover classification, is used to mask pixels categorized as "No Data", "Dark Area Pixels", "Clouds high probability", or "Cirrus". However, the category "Snow and Ice" often represents whitewater in coastal zones, and is therefore deliberately not masked.

In line with well-established approaches (Vos et al., 2023b), coastal waterlines are mapped by applying Otsu-thresholding to mNDWI optical satellite imagery and extracted at sub-pixel resolution using marching-squares. The Otsu-threshold value is computed after applying a pixel classification, to consider exclusively sandy and water pixels in the thresholding process. Additionally, we add a simple quality control filter to exclude imagery with over 95% water pixels in the S2 SCL or CoastSat classification, as such high water content does not yield effective results when using the Otsu' algorithm. Finally, by intelligently designing the computational graph, we simultaneously compute two results: a raster map with the coastal water occurrence probability, as the mean presence of water pixels over time; and a vector layer with waterlines from each image. We emphasize that this experiment does not aim to introduce yet another method for shoreline monitoring, but rather serves as relatable benchmark to the efficiency of data-proximate cloud computing for coastal monitoring.

2.8.2. Experiment 2: Geospatial data retrieval

This experiment benchmarks the efficiency of geospatial data retrieval using eight different data dissemination strategies by retrieving coastal transects for the Basque Country, Spain, from the GCTS. We evaluate the efficiency gains of cloud-optimized data, geospatial sorting, and metadata filtering with STAC across different data models (GeoPandas, Dask GeoPandas, and DuckDB) and retrieval methods (spatial join and predicate pushdown). Geospatial sorting, which involves sorting data based on quadkey (a geohash facilitating efficient spatial indexing, also see 2.5), is examined. Metadata filtering, performed on the attributes provided in the STAC collection, allows for selective retrieval of relevant data partitions. We also compare retrieval methods, including spatial join operations, which merge datasets based on their spatial relationship, and predicate pushdown, a query optimization technique that applies filters early in the data retrieval process to enhance performance by reducing the amount of data transferred and processed.

2.8.3. Case study: Coastal elevation mapping

This case study demonstrates the potential of cloud-native workflows for coastal science by combining data-proximate computing (Expt. 1; Section 2.8.1) with cloud-optimized data accessed via a standardized metadata specification (Expt. 2; Section 2.3). Specifically, we integrate the GCTS with DeltaDTM, a novel digital terrain model (Pronk et al., 2024), to determine the percentage of the world's first kilometer of coast that is lower than 5 m. The GCTS consists of more than 11 million coastal transects, while DeltaDTM includes 7105 tiles of 1 x 1 degree each at a spatial resolution of one arc second (30 m). We migrated DeltaDTM to cloud object storage and described it in a STAC catalog for convenient analysis. Thus, for this analysis both datasets are stored in cloud-optimized formats and accessible using STAC into a cloud-based computational cluster located close to data storage. DeltaDTM tiles are grouped by quadkey at zoom level 4, and relevant transects from the GCTS are retrieved using Dask GeoPandas, with STAC effectively filtering the necessary partitions. The transect data is then broadcast across the client, allowing it to be referenced by pointers without transmitting the data over the network, ensuring efficient processing. This setup establishes an efficient, scalable data processing routine capable of high-resolution extraction on a global scale.

3. Results

3.1. Where should we run our algorithms?

We analyzed the efficiency and scalability of data-proximate, cloud-based coastal waterline mapping. The results demonstrate that mapping shorelines in close proximity to where the satellite data is stored is 70 times faster for small areas, such as at Narrabeen Beach, Australia (Appendix A.2). This is a significant improvement that highlights the efficiency gains of data-proximate computing, which are specifically relevant for coastal monitoring because this practice requires frequent data processing to track dynamic shoreline changes as satellite imagery comes available. Moreover, a comparison (Appendix A.1) shows that this cloud-native CoastPy approach produces results that are in good agreement with CoastSat, making it an important step towards instantaneous shoreline mapping at extensive spatial scales. Finally, data-proximate coastal monitoring is more sustainable as it eliminates the need to duplicate or move large datasets and efficiently utilizes on-demand compute resources.

Another key result relates to scalability. We assessed the scalability of our cloud-native approach to coastal waterline mapping by expanding the study area to regional and state levels, first examining the San Francisco Peninsula and then the entire state of California. While scaling up to larger areas (e.g., Ocean Beach) the computation becomes relatively more efficient (Appendix A.3), with CPU usage almost fully saturated at 100% while operating at scale. While processing areas as large as Sentinel-2 tiles (110 x 110 km) exceeded the memory capacity of our Dask workers (8 GB RAM per worker), the workflow remained efficient at the California state level when tiles were subdivided into zoom-level 10 quadtiles (38 x 38 km) (See Section 2.5). The results, as shown in Fig. 2, indicate that coastal areas spanning quadtiles at zoom level 10 can be processed in approximately five minutes on average; that equivalents to coastal waterline mapping at 50 km per seconds, which is several orders of magnitude faster than conventional approaches, such as CoastSat (0.1 km/s; Appendix A.2). Although we find that occasionally data is spilled to disk during the operation, reporting double wall-clock times (10 min), the cluster recovers to a healthy state, demonstrating the robustness of this approach. This improved efficiency at larger scales is because proportionally less time spent configuring the workers and constructing the computational graph (Appendix A.3). Overall, these findings highlight that specialized coastal monitoring routines can be much more efficiently orchestrated in the cloud, close to where the data is physically stored, with increased efficiency as the study area scales from local to regional levels.

Table 1
Average execution time (s) for various strategies of data retrieval. The crosses (X) indicate whether a certain strategy was applied.

Experiment	Data model	Query method	Cloud-optimized	Spatial sort	STAC filter	mean (s)	std (s)
Expt. 1	GeoPandas	Spatial join				1071.3	5.0
Expt. 2	Dask GeoPandas	Spatial join	X	X		41.3	1.1
Expt. 3	Dask GeoPandas	Spatial join	X			40.5	1.0
Expt. 4	DuckDB	Spatial join	X	X		25.2	1.1
Expt. 5	DuckDB	Spatial join	X			24.0	0.8
Expt. 6	Dask GeoPandas	Spatial join	X	X	X	10.9	0.3
Expt. 7	DuckDB	Predicate pushdown	X	X		7.4	0.6
Expt. 8	DuckDB	Predicate pushdown	X	X	X	6.7	0.7

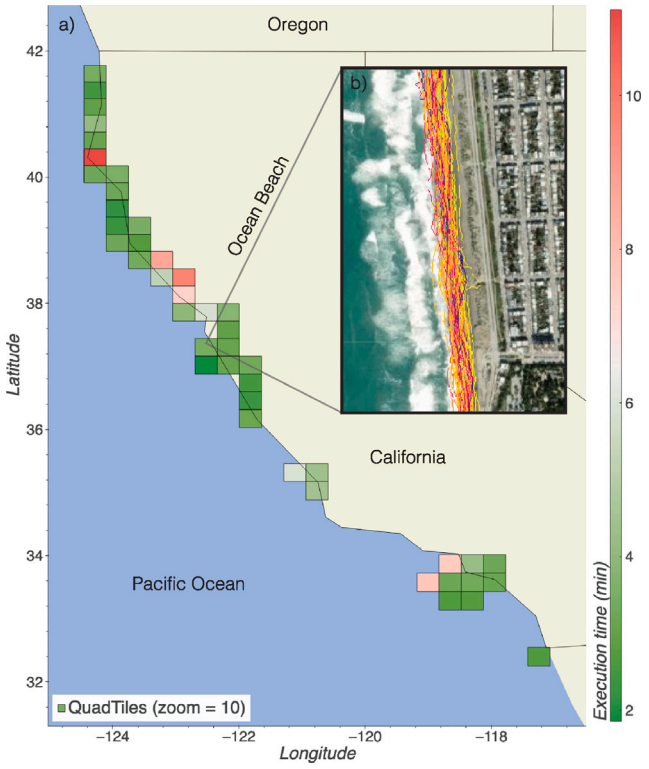


Fig. 2. Execution time (min) per quadtile for cloud-native coastal waterline mapping across California state, USA. By orchestrating shoreline-monitoring routines in close proximity of the satellite data, coastal waterlines are mapped at approximately 50 km/s.

3.2. How should we store our data?

We benchmarked eight different geospatial data retrieval strategies. Table 1 summarizes the characteristics and performance of each strategy, including mean and standard deviation of execution times over 20 iterations. Detailed distributions of retrieval times are presented in Appendix C.

In the initial experiment, as presented in Table 1, coastal transects were retrieved from the GCTS, that is stored as a traditional geopackage, which is approximately 5 GB in that format. Unlike in other experiments, this retrieval operation had to be conducted on a personal computer due to memory limitations on a standard MSPC Pangeo instance. This setup resulted in data retrieval that is up to 160 times slower (Expt. 1 vs. Expt. 8) compared to those achieved with cloud-optimized data formats. Thus, the occurrence of memory errors and the prolonged execution time for data retrieval in this experiment show the convenience and advantages of cloud-optimized data formats.

Subsequent experiments involve fetching transects from the GCTS stored across several GeoParquet partitions, that are altogether approximately 1 GB in this cloud-optimized format. The experiments use various techniques such as spatial joins or predicate pushdown (See Section 2.8.2), facilitated by a STAC catalog to identify relevant partitions and spatial sorting by quadkey. Surprisingly, the results (Table 1) indicate that spatial sorting slightly increases data retrieval times. However, the use of STAC metadata to effectively filter the relevant STAC items enhances Dask GeoPandas so that it is capable of retrieving transects four times faster (Expt. 4 vs. Expt 6), likely due to the reduced need to index data objects using HTTP protocol. This efficiency gain demonstrates the advantage of standardized metadata in optimizing data retrieval, while also underscoring the importance of geospatial sorting, as the effectiveness of the STAC metadata filter depends on it.

The results (Table 1) demonstrate that the DuckDB query engine is more efficient for geospatial data retrieval than Dask-GeoPandas (e.g., Expt 4 vs 2) in this setup, being almost twice as fast. More importantly, the data show that employing predicate pushdown on a bounding box attribute is a three times more efficient strategy for data retrieval than a spatial join operation (Expt 7 vs. Expt 5). For geospatial predicate pushdown to function effectively, the data must be geospatially sorted and include a bounding box column that provides the extent of the geometry. This underscores the importance of adding a bounding box column to geometries and organizing the data geospatially to significantly enhance query performance. Additionally, the data retrieval is slightly faster (10%) when using STAC metadata to selectively filter relevant STAC items (e.g., Expt. 8 vs. Expt. 7), although the efficiency gains from this approach do not match those achieved with Dask GeoPandas (twice as fast). While the differences in execution times among strategies that access cloud-optimized data formats are of a different order than the 160 times improvement reported against conventional data formats, the differences are relevant for analysis at scale. Implementing simple geospatial sorting methods combined with standardized metadata specifications results in approximately 4 times faster performance. This is particularly important when these access patterns are repeated for a large number of similar tasks, demonstrating the effectiveness of these strategies in optimizing geospatial data retrieval at scale.

In summary, these experiments affirm the importance of cloud-optimized data formats for broad-scale coastal analytics, where data interoperability is crucial (Section 1). Cloud-optimized data formats, when combined with geospatial sorting and query optimization techniques like predicate pushdown, facilitate efficient filtering methods for retrieving data pertinent to specific regions of interest. Additionally, this experiment highlights the value of metadata-driven data access strategies in enhancing the efficiency of data retrieval, which is presumably increasingly important as the number of partitions grows and indexing remote data objects over HTTP protocol becomes a bottleneck. Therefore, adopting cloud-optimized data formats, geospatial sorting, and enabling metadata-driven access methods is essential for facilitating efficient geospatial data retrieval.

3.3. How much of the first km of coastal land is below 5 m?

By using cloud technology as described in earlier experiments (See Sections 3.1 and 3.2) we developed a scalable, high-resolution mapping at global scale. Fig. 3 illustrates the fine spatial resolution at which elevation data is extracted along the landward side of the transects

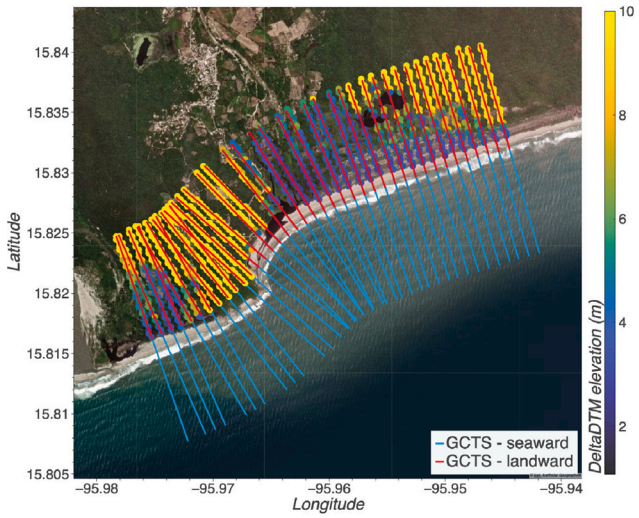


Fig. 3. Extracting DeltaDTM elevation data over the landward side (1 km) of our coastal transects, around Barra de la Cruz, Mexico.

around Barra de la Cruz, Mexico. In total we extract approximately 300 million elevation observations at equally-spaced 100 m alongshore resolution. Compared to the conventional download-and-analyze approach, this method significantly enhances our ability to analyze coastal datasets at global scale. We have developed a scalable method to locate and process DeltaDTM tiles. Similarly, coastal transects for a given tile can be retrieved downloading and processing an extensive 5 GB GeoPackage. A traditional workflow, involving a spatial join for each of the 7105 tiles, would take approximately 88 days to process just to retrieve the transects. In contrast, using cloud technology, we completed the entire analysis in a few hours, making it 700 times faster.

On average, we find that 33% of the first km in the coastal zone is lower than 5 meter. Fig. 4 shows the average percentage of very low-lying coastal land (< 5 m) in the first km coastal zone, with the 5% (n=25) largest clusters (Anselin, 1995) indicated as red dots. The map shows that particularly the gulf of Mexico, the East Coast of the United States and the European Wadden Sea much of the first km of coastal land is lower than 5 m above mean sea level. In this 1-km coastal zone, the distribution (Fig. 4) of areas that are on average lower/higher than 5 m is highly bi-modal; which shows that either the 1-km coastal land is mostly below 5 m or, land rises fast and elevation is mostly above 5 m.

4. Discussion

Cloud technology and the opening up of EO data have started a digital transformation in coastal science. In the introduction, we distinguished between two distinct analysis strategies: one aims for global coverage, often compromising accuracy for spatial extent (“everywhere”), while the other prioritizes accuracy (“anywhere”). This paper demonstrates how leveraging an open, flexible, and scalable geospatial software stack (Pangeo) in combination with cloud-optimized data formats bridges the gap between these approaches, enabling coastal analytics at broad scales without compromising the high spatio-temporal resolution and accuracy that are typically used in more local analyses.

An essential step this approach is decoupling data storage from compute, a strategy that contrasts with the integrated frameworks of existing platforms like GEE. This separation provides more control over both processes, enabling independent management of each.

Compute, is managed on an open, flexible and scalable framework (Pangeo), that, depending on the needs of the analysis, can be deployed on personal computers, HPC, or cloud infrastructure. As shown in the first experiment, this software ecosystem provides a flexible environment that can be scaled up to broader areas while retaining the spatio-temporal resolution used in some specialized coastal monitoring practices.

Storage, in turn, is handled using cloud-optimized data formats, while adhering to standardized metadata specifications. As shown in the second experiment, this structured approach enables rapid querying, and ensures that data is optimized for coastal analytics at scale.

Implementing and maintaining such a cloud solution presents some technical challenges. Besides writing scalable software, producing cloud-based data repositories requires expertise in cloud-optimized data formats, data specifications and data partitioning. While such skills may not be commonly expected from coastal scientists, our case study demonstrates that the substantial improvements in efficiency, flexibility, and hence analytical capabilities justify these efforts.

The modularity of cloud-based coastal analytics eases the migration between different computational environments while removing dependency on proprietary platforms (e.g., GEE). Also, by decoupling storage and compute, this system can continuously integrate the latest innovations in both data management and algorithms. Other advantages of cloud-based solutions is that it provides universal access points to data, that are much more accessible than traditional storage solutions, while they typically also reduce the need for data downloading with in effect less data duplication.

Cloud computing has a significant environmental impact (Monserrate, 2022), and data-intensive applications — particularly those utilizing Artificial Intelligence (AI) — are a major source of its energy consumption (Katal et al., 2023). Coastal research labs that leverage this powerful infrastructure should be mindful of their environmental footprint, especially as their funding is often tied to initiatives promoting sustainability and/or addressing climate change. However, despite its notable carbon footprint, cloud computing can offer more sustainable solutions (Jones, 2018). By optimizing resource allocation, reducing idle time, and consolidating large-scale operations, cloud systems can minimize overall energy consumption. Additionally, cloud infrastructure’s universal access points reduce the need for data duplication and unnecessary transfers, while cloud-optimized data formats, through efficient compression, further minimize storage and transmission requirements, enhancing overall sustainability.

Although there are technical barriers and environmental impacts that require careful attention, the importance of cloud technology for large-scale coastal analytics is clear, as also shown in the experimental findings, that are discussed in the next section.

4.1. Experimental findings

In this section we discuss several key results from our experiments, beginning with the novel global coastal transect system (GCTS). The transect system benefits from recent OSM data uptake, corrects zonal bias, includes polar latitudes, and offers a finer alongshore resolution (100 m) than existing transect systems. We believe this transect system can serve as a robust foundation for various coastal studies, providing a reliable basis for deriving SDS series, coastal characterizations, classifications, and related statistics.

Secondly, we show that 33% of the first kilometer of coast is below 5 m. This data, calculated at a 100-m alongshore resolution, is crucial for several kinds of coastal analyses, such as coastal classification and characterizations. This finding is particularly relevant in the context of the Low Elevation Coastal Zone (LECZ), as described by McGranahan et al. (2007), potentially providing a more detailed understanding of coasts that are vulnerable to accelerating climate change. However, the 30-m resolution of the Digital Terrain Model (DTM) is likely insufficient for flood-risk modeling, underscoring the continued relevance of local studies. The workflows used can be adopted to map other coastal variables onto, for example, our transect system (GCTS). With more coastal variables mapped, it will become possible to apply theoretical classification frameworks (e.g., Cooper and McLaughlin, 1998; Finkl,

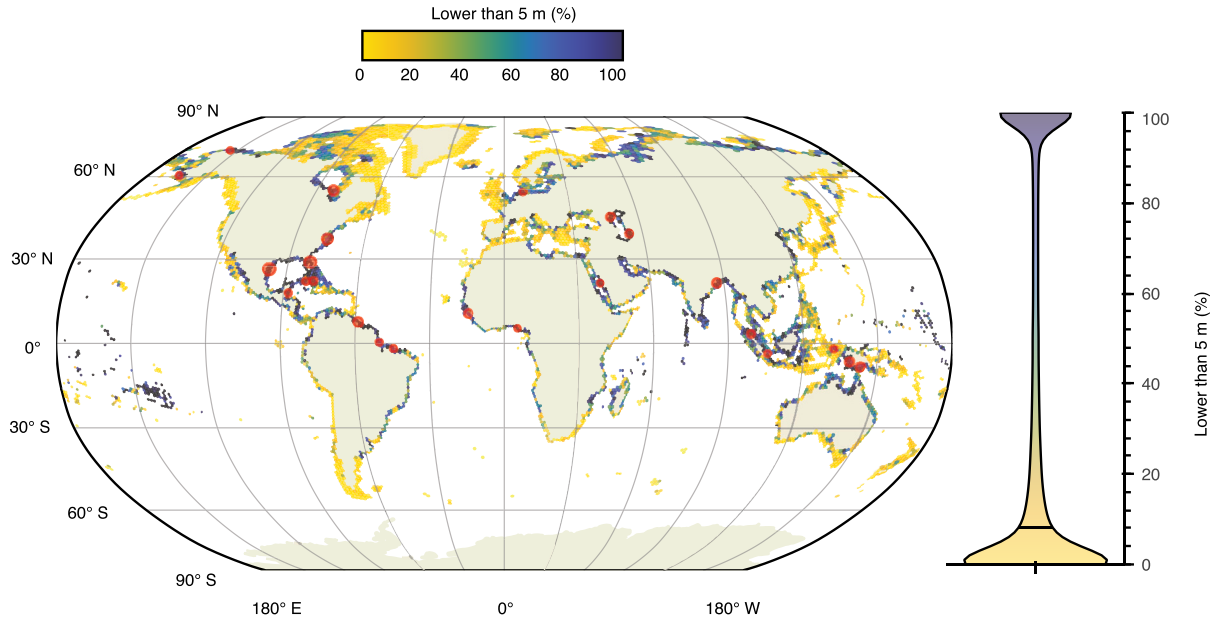


Fig. 4. The map shows the average percentage of land within the first kilometer of the coastal zone that is lower than 5 m above mean sea level, based on approximately 300 million observations from DeltaDTM (Pronk et al., 2024) across more than 11 million transects (GCTS). On average, 33% of this coastal zone is below 5 m. Red dots indicate the 5% largest clusters of predominantly low-lying coastal land. To the right, a violin plot shows the distribution of the percentage of coastal land below 5 m.

2004) at extensive scales without compromising spatial resolution and accuracy.

Thirdly, we present open, scalable, flexible methods for efficient coastal analytics at planetary scale, that are up to 700 times faster than traditional download-analyze approaches. This tremendous speed-up is achieved by bringing code to the data rather than moving the data to the code; and, using cloud-optimized data exposed through rich, standardized metadata specifications. By adopting this framework for coastal waterline mapping, we achieved a processing speed of 50 km/s, compared to 0.1 km/s using conventional methods, like CoastSat. Data-proximate computing, increasingly standard in various scientific fields (Gentemann et al., 2021), is very relevant for the coastal community, which increasingly relies on EO data (Vitousek et al., 2023a), that is still typically downloaded and analyzed on institutional premises.

The coastal waterlines mapped during the data-proximate shoreline monitoring experiment are primarily intended to demonstrate the feasibility of efficiently implementing specialized coastal monitoring routines in the cloud, rather than serving as a dataset for studying coastal dynamics. Nevertheless, the methods introduced here have the potential to form the basis for the first-ever global mapping of instantaneous shorelines from the full historical Landsat and/or Sentinel-2 catalog. Crucially, future enhancements must include tidal corrections and improvements to the classification methods to ensure robust generalization across diverse coastal environments (Vos et al., 2023b; Konstantinou et al., 2023), including macro-tidal regions and beaches with unique sand types, like those on volcanic islands.

Fourthly, we show that for coastal analytics at scale, cloud-optimized data exposed through standardized metadata specifications are essential. Our experiments on geospatial data retrieval indicate that data storage format and accompanied metadata are as important as the data itself, especially for large-scale analyses. By eliminating repetitive tasks such as locating data for regions of interest and enhancing data interoperability, we developed data retrieval methods that are up to 160 times faster than traditional download-and-analyze approaches. Another advantage is that by adopting these principles, the data de facto becomes Findable, Accessible, Interoperable and Reusable (FAIR) (Wilkinson et al., 2016).

A crucial step in curating the data is partitioning. In this study, we explored various partitioning methods, with our released GCTS partitioned by quadkey to enhance the efficiency of regional data access and time-series analysis. While partitioning based on administrative boundaries has been suggested (Holmes, 2023), it poses challenges in coastal science. Administrative boundaries can enhance user-friendliness, but the coast — being a transitional area between ocean and land — is not always fully encompassed by existing administrative divisions. Additionally, the complexity and variability of coastal regions can result in highly unequal partitions, further complicating this approach.

Our geospatial partitioning strategy, while effective, is not definitive, especially for datasets that grow with time, such as with ongoing satellite missions. Managing spatio-temporal coastal data repositories, such as SDS, presents significant challenges due to conflicting partitioning needs. Spatial partitioning enables efficient regional data access, that is useful for time-series analysis, but requires frequent data rewriting, which is computationally intensive and costly, particularly in cloud-object stores. As a result, repositories may be managed through temporal partitioning, where data is appended as new satellite data becomes available. While this is more logical for continuously updating datasets, it is less efficient for users needing access to time series of certain coastal variables per region.

Fifthly, the cloud-native coastal waterline mapping experiment demonstrates that we can now incorporate advanced ML models into coastal analyses at scale. Although the classification model used in this study is relatively basic (a simple feed-forward network), it serves as proof of concept that advanced coastal deep learning models can be integrated into server-side computing. This capability is critical for implementing advanced coastal ML models (e.g., Buscombe and Ritchie, 2018; Al Najar et al., 2023) at extensive spatial scales.

4.2. Outlook

We argue that the coastal community should collectively begin constructing publicly available, analysis-ready coastal data repositories, as these will be the critical resources for powering coastal ML and providing a robust foundation for data-driven coastal decision making. The digital revolution in coastal science, triggered by the opening up of historical satellite catalogs and innovations in cloud technology, has provided an unprecedented global perspective of the coastal environment. However, the coastal community is not yet fully capitalizing on recent advances in ML, such as Deep Learning (DL).

Although the ongoing digital revolution in coastal science is most likely going to culminate in the widespread adoption of coastal AI, this advancement requires the availability of high-quality data. Adopting best practices from computer science (Raymond, 1999) and making data publicly available (Tenopir et al., 2015) is crucial for achieving this goal, but not enough. Now that coastal science has an appetite for data, its management must be recognized as an integral component of coastal research. Without a focus on data management, we risk accumulating more data than we can effectively analyze. Future coastal data releases should aim to minimize manual, repetitive work in downstream coastal analytics, by providing analysis-ready data, following flexible data schemata that we, as a coastal community, still have to work out ourselves. Yet it is also critical that future coastal analytics integrate strategies to reduce environmental footprints, particularly for the data-intensive applications. To ensure the long-term sustainability and accessibility of the data resources as well as the tools to work with it, governmental support (e.g., Directorate-General for Research and Innovation (European Commission), 2022) to develop and maintain public cloud infrastructure is essential, as reliance solely on commercial enterprises may compromise the stability and equitable access to potentially critical coastal data. We envision a future where an Earth System data cube (Mahecha et al., 2020), enriched with coastal characteristics, will allow customized views of the coast, possibly even through natural language queries (Zhu et al., 2024) enabled by artificial intelligence.

5. Conclusion

This study demonstrates the transformative potential of cloud technology for coastal analytics at extensive spatial scales. We introduced a novel global coastal transect system (GCTS), found that 33% of the first kilometer of coast is below 5 m, and developed methods that are up to 700 times faster than conventional approaches. The GCTS introduced here can serve as a foundational coastal dataset, offering a robust frame of reference with a global set of coastal stations that can be used for deriving shoreline-change series, coastal characterizations, and related statistics. Our findings highlight that coastal science no longer needs to be constrained by high latency, storage capacity, available compute resources, or specific toolboxes provided by cloud platforms. By leveraging cloud technology and a flexible, scalable software ecosystem, we can perform complex computations close to data storage, drastically reducing analysis time from months to just a few hours. This approach allows us to use all available data without compromising accuracy or resolution, hopefully setting a precedent for future broad-scale coastal studies. We are therefore convinced that if the coastal community aims to address urgent coastal challenges at broad spatial scale without compromising accuracy or spatio-temporal resolution — bridging the gap from "everywhere" towards "anywhere" — it will have to start producing tools, models, and data suitable for scalable coastal analytics.

Software availability

Software name: CoastPy

Developer: Floris Calkoen

Year first official release: 2024

System requirements: Mac, Linux, Windows

Program language: Python

Program size: <1 MB

Availability: https://github.com/TUDelft-CITG/coastpy

License: MIT

Documentation: README, documentation and tutorial notebooks.

Citation: This paper.

CRediT authorship contribution statement

Floris Reinier Calkoen: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Arjen Pieter Luijendijk:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Kilian Vos:** Writing – review & editing, Validation, Investigation. **Etiënne Kras:** Writing – review & editing, Visualization, Software, Data curation. **Fedor Baart:** Writing – review & editing, Supervision, Resources, Methodology, Conceptualization.

Acknowledgements

We would like to thank Reviewer 1, Reviewer 2 and Reviewer 3 for their invaluable comments; particularly because they have helped to better communicate our key findings while also placing our work in a broader coastal context.

This research was funded by European Commission SOCIETAL CHALLENGES - Climate action, Environment, Resource Efficiency and Raw Materials as part of CoCliCo (Grant agreement ID: 101003598); and Deltares strategic research programs Moonshot 2 on Flooding and Enabling Technologies.

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.

Appendix A. Shoreline monitoring

This appendix provides supplementary material supporting the findings of data-proximate coastal waterline mapping, including data performance as well as qualitative comparison.

A.1. Qualitative comparison between CoastSat and CoastPy

Fig. 5 presents time series data for a transect (PF6) at Narrabeen Beach, Australia, comparing field observations, CoastSat, and CoastPy. The figure demonstrates that CoastPy achieves comparable accuracy in mapping coastal waterlines to CoastSat. The minor discrepancies between CoastSat and CoastPy can be attributed to the tidal correction applied in CoastSat but not yet implemented in CoastPy. These results confirm that the methods used in this cloud-native coastal monitoring approach are effectively configured.

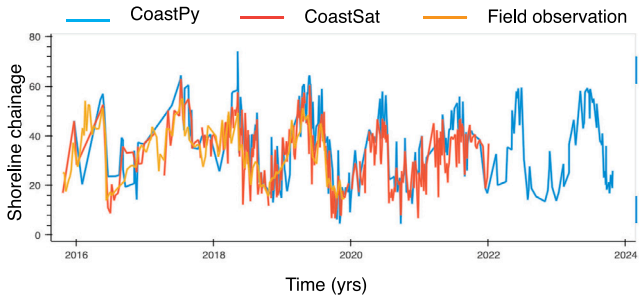


Fig. 5. Comparison of CoastPy, CoastSat, and field-observation SDS-series for transect PF6 at Narrabeen Beach, Australia.

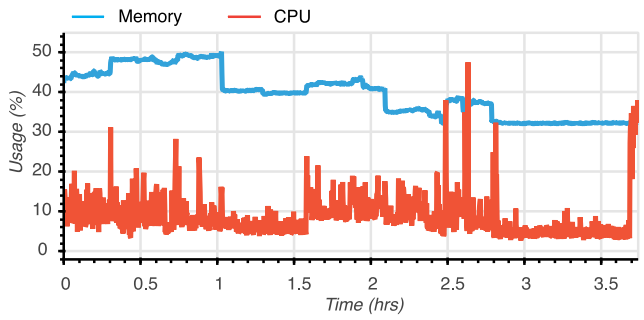


Fig. 6. CPU and memory usage over time for CoastSat while mapping SDS from the full Sentinel-2 catalog for Narrabeen Beach, Australia.

A.2. Computational performance CoastSat

Fig. 6 shows time-series of memory and CPU usage while mapping SDS for Narrabeen Beach, Australia, using CoastSat. The mapping of all shorelines from the Sentinel-2 catalog took 3.75 h. During this period, memory usage was saturated at approximately 50% on average, while CPU usage averaged around 15%. In total, 851 Sentinel-2 Top of Atmosphere (TOA) images were processed, resulting in 450 unique shorelines after accounting for duplicates due to overlapping tiles.

A.3. Computational performance CoastPy

Fig. 7 presents a performance report with CPU, memory, and bandwidth usage of the Dask compute cluster while mapping coastal waterlines using CoastPy for Ocean Beach, USA. The area of interest covers two quadtiles (zoom level 10), necessitating two iterations of mapping, which is evident from the repetitive pattern in the data over time. Each tile mapping follows a repetitive pattern of scheduling, initial computation with few workers, adaptive scaling, and computation with many workers. During the scheduling phase, CPU, memory, and bandwidth usage are low. As the computation begins with a small number of workers, CPU usage increases until the cluster adaptively scales to more workers to handle the larger computational workload. The adaptive scaling phase shows relatively low CPU usage as workers are configured with the necessary software and data. The compute-at-scale phase occupies approximately one-fourth of the total time, similar to each of the other phases, and is characterized by high bandwidth usage (intensive write operations), showing that the majority of coastal waterlines are mapped during this phase. This cloud-native approach to coastal waterline mapping achieves near-full CPU saturation when computing results.

Appendix B. Geospatial data partitioning

Fig. 8 presents a global map of the GCTS, spatially partitioned into parts, each containing a maximum size of 100 MB. The data is sorted by quadkey, with entries from Alaska at the beginning and entries from Australia at the end. Bounding boxes representing the different partitions are overlaid on the map, showing that transect records are grouped by spatial area. This partitioning approach facilitates efficient geospatial data retrieval by allowing metadata to discard irrelevant partitions. The figure also shows that areas of higher coastal complexity, such as Indonesia, have relatively high data density.

Appendix C. Data retrieval

Fig. 9 presents the distribution of data retrieval times for various data release strategies, as detailed in Table 1. The small variance across all experiments indicates a robust setup, ensuring consistent performance across different retrieval methods.

Data availability

The authors share their code and instructions to access the data at https://github.com/TUDelft-CITG/coastpy. The Global Coastal Transect System is also available for download at https://zenodo.org/records/14056925.

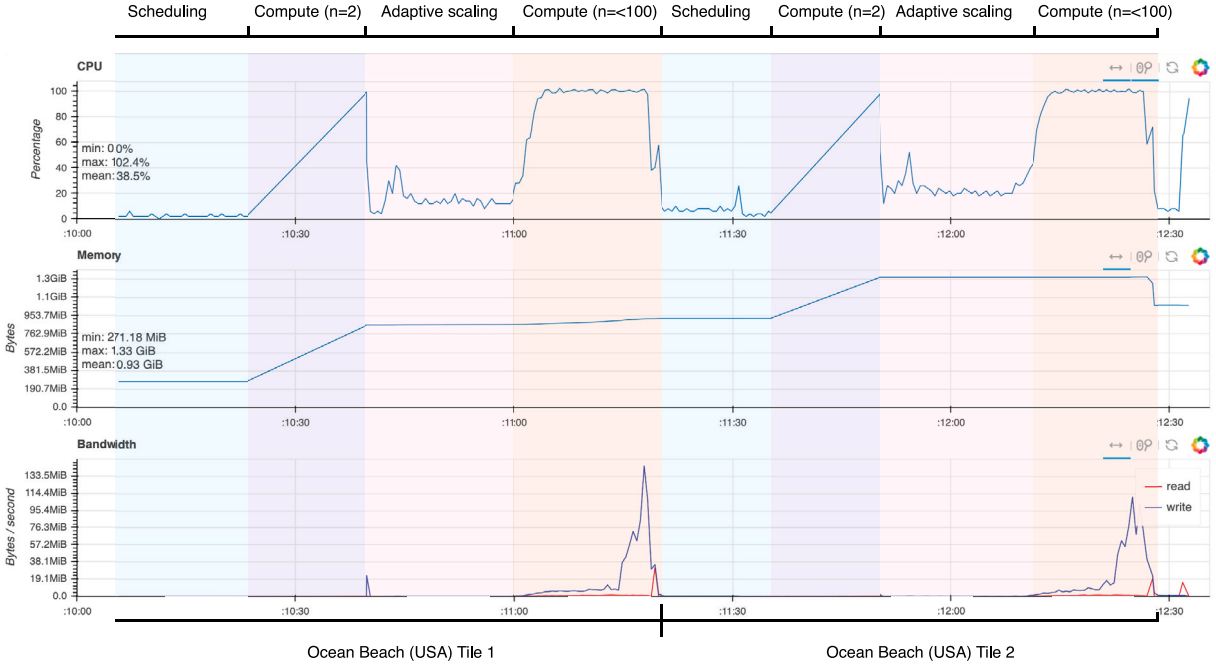


Fig. 7. Performance report of the Dask compute cluster during coastal waterline mapping using CoastPy for Ocean Beach, USA. The report details CPU, memory, and bandwidth usage over time, highlighting the repetitive patterns of scheduling, initial computation, adaptive scaling, and compute-at-scale phases for two quadtiles.

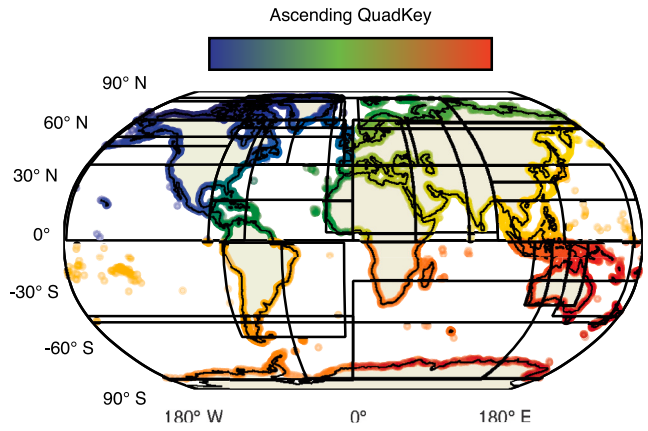


Fig. 8. Global map of the geospatially partitioned GCTS data, sorted by quadkey. The dataset is divided into partitions with a maximum size of 100 MB, enabling efficient geospatial data retrieval.

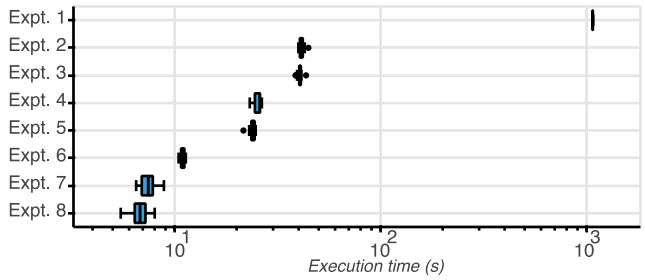


Fig. 9. Distribution of data retrieval times for different data release strategies. The specific operations applied in each experiment are detailed in Table 1.

References

Abernathey, R.P., Augspurger, T., Banihirwe, A., Blackmon-Luca, C.C., Crone, T.J., Gentemann, C.L., Hamman, J.J., Henderson, N., Lepore, C., McCaie, T.A., Robinson, N.H., Signell, R.P., 2021. Cloud-native repositories for big scientific data. Comput. Sci. Eng. 23 (2), 26–35. http://dx.doi.org/10.1109/MCSE.2021.3059437.

Abernathey, R., Paul, K., Hamman, J., Rocklin, M., Lepore, C., Tippett, M., Henderson, N., Seager, R., May, R., Del Vento, D., 2017. Pangeo NSF earthcube proposal. Figshare.

Al Najar, M., Thoumyre, G., Bergsma, E.W.J., Almar, R., Benshila, R., Wilson, D.G., 2023. Satellite derived bathymetry using deep learning. Mach. Learn. 112 (4), 1107–1130. http://dx.doi.org/10.1007/s10994-021-05977-w.

Anselin, L., 1995. Local indicators of spatial association—LISA. Geogr. Anal. 27 (2), 93–115. http://dx.doi.org/10.1111/j.1538-4632.1995.tb00338.x.

Bauer-Marschallinger, B., Falkner, K., 2023. Wasting petabytes: A survey of the sentinel-2 UTM tiling grid and its spatial overhead. ISPRS J. Photogramm. Remote Sens. 202, 682–690. http://dx.doi.org/10.1016/j.isprsjprs.2023.07.015.

Baumann, P., 1993. Language support for raster image manipulation in databases. In: Göbel, M., Teixeira, J.C. (Eds.), Graphics Modeling and Visualization in Science and Technology. Springer, Berlin, Heidelberg, pp. 236–245. http://dx.doi.org/10.1007/978-3-642-77811-7_19.

Bishop-Taylor, R., Nanson, R., Sagar, S., Lymburner, L., 2021. Mapping Australia's dynamic coastline at mean sea level using three decades of landsat imagery. Remote Sens. Environ. 267, 112734. http://dx.doi.org/10.1016/j.rse.2021.112734.

Buscombe, D., Ritchie, A.C., 2018. Landscape classification with deep neural networks. Geosciences 8 (7), 244. http://dx.doi.org/10.3390/geosciences8070244.

Castelle, B., Kras, E., Masselink, G., Scott, T., Konstantinou, A., Luijendijk, A., 2024. Satellite-derived sandy shoreline trends and interannual variability along the Atlantic coast of Europe. Sci. Rep. 14 (1), 13002. http://dx.doi.org/10.1038/s41598-024-63849-4.

Cooper, J.A.G., McLaughlin, S., 1998. Contemporary multidisciplinary approaches to coastal classification and environmental risk analysis. J. Coast. Res. 14 (2), 512–524, arXiv:4298806.

Cornillon, P., Gallagher, J., Sgouros, T., 2003. OPeNDAP: Accessing data in a distributed, heterogeneous environment. Data Sci. J. 2, 164–174. http://dx.doi.org/10.2481/dsj.2.164.

Dean, J., Ghemawat, S., 2008. MapReduce: Simplified data processing on large clusters. Commun. ACM 51 (1), 107–113. http://dx.doi.org/10.1145/1327452.1327492.

Directorate-General for Research and Innovation (European Commission), 2022. The Digital Twin Ocean: An Interactive Replica of the Ocean for Better Decision Making. Publications Office of the European Union.

Durbin, C., Quinn, P., Shum, D., 2020. Task 51 - cloud-optimized format study.

Finkl, C.W., 2004. Coastal classification: Systematic approaches to consider in the development of a comprehensive scheme. J. Coast. Res. 20 (1), 166–213. http://dx.doi.org/10.2112/1551-5036(2004)20[166:CCSATC]2.0.CO;2.

Gavin, D., Dhu, T., Sagar, S., Mueller, N., Dunn, B., Lewis, A., Lymburner, L., Minchin, S., Oliver, S., Ross, J., Thankappan, M., 2018. Digital earth Australia - from satellite data to better decisions. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, Valencia, pp. 8633–8635. http://dx.doi.org/10.1109/IGARSS.2018.8518160.

Gentemann, C.L., Holdgraf, C., Abernathey, R., Crichton, D., Colliander, J., Kearns, E.J., Panda, Y., Signell, R.P., 2021. Science storms the cloud. AGU Advances 2 (2), e2020AV000354. http://dx.doi.org/10.1029/2020AV000354.

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.

Holmes, C., 2023. The admin-partitioned GeoParquet distribution.

Hormann, C., 2014. Generalisierung im Raster für Karten kleiner Maßstäbe — mit Anwendungsbeispielen aus OpenStreetMap. KN - J. Cartogr. Geogr. Inform. 64 (5), 276–280. http://dx.doi.org/10.1007/BF03544188.

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

Hulskamp, R., Luijendijk, A., van Maren, B., Moreno-Rodenas, A., Calkoen, F., Kras, E., Lhermitte, S., Aarninkhof, S., 2023. Global distribution and dynamics of Muddy Coasts. Nature Commun. 14 (1), 8259. http://dx.doi.org/10.1038/s41467-023-43819-6.

Jones, N., 2018. How to stop data centres from gobbling up the world's electricity. Nature 561 (7722), 163–166. http://dx.doi.org/10.1038/d41586-018-06610-y.

Katal, A., Dahiya, S., Choudhury, T., 2023. Energy efficiency in cloud computing data centers: A survey on software technologies. Cluster Comput. 26 (3), 1845–1875. http://dx.doi.org/10.1007/s10586-022-03713-0.

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.

Konstantinou, A., Scott, T., Masselink, G., Stokes, K., Conley, D., Castelle, B., 2023. Satellite-based shoreline detection along high-energy Macrotidal coasts and influence of beach state. Mar. Geol. 462, 107082. http://dx.doi.org/10.1016/j.margeo.2023.107082.

Luijendijk, A., Hagenaars, G., Ranasinghe, R., Baart, F., Donchyts, G., Aarninkhof, S., 2018. The state of the world's beaches. Sci. Rep. 8 (1), 6641. http://dx.doi.org/10.1038/s41598-018-24630-6.

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.

Mao, Y., Harris, D.L., Xie, Z., Phinn, S., 2021. Efficient measurement of large-scale decadal shoreline change with increased accuracy in tide-Dominated coastal environments with google earth engine. ISPRS J. Photogramm. Remote Sens. 181, 385–399. http://dx.doi.org/10.1016/j.isprsjprs.2021.09.021.

McGranahan, G., Balk, D., Anderson, B., 2007. The rising tide: Assessing the risks of climate change and human settlements in low elevation coastal zones. Environ. Urbanization 19 (1), 17–37. http://dx.doi.org/10.1177/0956247807076960.

McKinney, W., 2010. Data structures for statistical computing in Python. In: Python in Science Conference. Austin, Texas, pp. 56–61. http://dx.doi.org/10.25080/Majora-92bf1922-00a.

Medvedev, D., Lemson, G., Rippin, M., 2016. SciServer Compute: Bringing analysis close to the data. In: Proceedings of the 28th International Conference on Scientific and Statistical Database Management. In: SSDBM '16, Association for Computing Machinery, New York, NY, USA, pp. 1–4. http://dx.doi.org/10.1145/2949689.2949700.

Mikkelsen, A.B., McDonald, K.K., Kalksma, J., Tyrrell, Z.H., Fletcher, C.H., 2024. Three years of weekly DEMs, aerial orthomosaics and surveyed shoreline positions at Waikīkī Beach, Hawai'i. Sci. Data 11 (1), 324. http://dx.doi.org/10.1038/s41597-024-03160-z.

Monserrate, S.G., 2022. The cloud is material: On the environmental impacts of computation and data storage. MIT Case Stud. Soc. Ethical Responsib. Comput. (Winter 2022), http://dx.doi.org/10.21428/2c646de5.031d4553.

Muir, F.M.E., Hurst, M.D., Richardson-Foulger, L., Rennie, A.F., Naylor, L.A., 2024. VedgeSat: An automated, open-source toolkit for coastal change monitoring using satellite-derived vegetation edges. Earth Surf. Process. Landf. http://dx.doi.org/10.1002/esp.5835.

Murray, N.J., Phinn, S.R., DeWitt, M., Ferrari, R., Johnston, R., Lyons, M.B., Clinton, N., Thau, D., Fuller, R.A., 2018. The global distribution and trajectory of tidal flats. Nature 565 (7738), 222–225. http://dx.doi.org/10.1038/s41586-018-0805-8.

Pronk, M., Hooijer, A., Eilander, D., Haag, A., de Jong, T., Vousdoukas, M., Vernimmen, R., Ledoux, H., Eleveld, M., 2024. DeltaDTM: A global coastal digital terrain model. Sci. Data 11 (1), 273. http://dx.doi.org/10.1038/s41597-024-03091-9.

Raasveldt, M., Mühleisen, H., 2019. DuckDB: An embeddable analytical database. In: Proceedings of the 2019 International Conference on Management of Data. In: SIGMOD '19, Association for Computing Machinery, New York, NY, USA, pp. 1981–1984. http://dx.doi.org/10.1145/3299869.3320212.

Raoult, B., Bergeron, C., López Alós, A., Thépaut, J.N., Dee, D., 2017. Climate service develops user-friendly data store. ECMWF Newslett. Meteorology, 22–27. http://dx.doi.org/10.21957/P3C285.

Raymond, E., 1999. The cathedral and the bazaar. Knowl. Technol. Policy 12 (3), 23–49.

Rocklin, M., 2015. Dask: Parallel computation with blocked algorithms and task scheduling. In: Proceedings of the 14th Python in Science Conference. vol. 130, Citeseer, p. 136.

Roelvink, D., Huisman, B., Elghandour, A., Ghonim, M., Reyns, J., 2020. Efficient modeling of complex Sandy coastal evolution at monthly to century time scales. Front. Marine Sci. 7.

Tenopir, C., Dalton, E.D., Allard, S., Frame, M., Pjesivac, I., Birch, B., Pollock, D., Dorsett, K., 2015. Changes in data sharing and data reuse practices and perceptions among scientists worldwide. In: Van Den Besselaar, P. (Ed.), PLoS One 10 (8), e0134826. http://dx.doi.org/10.1371/journal.pone.0134826.

Vitousek, S., Buscombe, D., Vos, K., Barnard, P.L., Ritchie, A.C., Warrick, J.A., 2023a. The future of coastal monitoring through satellite remote sensing. Camb. Prisms Coast. Futures 1, e10. http://dx.doi.org/10.1017/cft.2022.4.

Vitousek, S., Vos, K., Splinter, K.D., Erikson, L., Barnard, P.L., 2023b. A model integrating satellite-derived shoreline observations for predicting fine-scale shoreline response to waves and sea-level rise across large coastal regions. http://dx.doi.org/10.22541/essoar.167839941.16313003/v1, Preprints.

Vos, K., Harley, M.D., Turner, I.L., Splinter, K.D., 2023a. Pacific shoreline erosion and accretion patterns controlled by El Niño/Southern Oscillation. Nat. Geosci. 16 (2), 140–146. http://dx.doi.org/10.1038/s41561-022-01117-8.

Vos, K., Splinter, K.D., Harley, M.D., Simmons, J.A., Turner, I.L., 2019. CoastSat: A google earth engine-enabled Python toolkit to extract shorelines from publicly available satellite imagery. Environ. Model. Softw. 122, 104528. http://dx.doi.org/10.1016/j.envsoft.2019.104528.

Vos, K., Splinter, K.D., Palomar-Vázquez, J., Pardo-Pascual, J.E., Almonacid-Caballer, J., Cabezas-Rabadán, C., Kras, E.C., Luijendijk, A.P., Calkoen, F., Almeida, L.P., Pais, D., Klein, A.H.F., Mao, Y., Harris, D., Castelle, B., Buscombe, D., Vitousek, S., 2023b. Benchmarking satellite-derived shoreline mapping algorithms. Commun. Earth Environ. 4 (1), 1–17. http://dx.doi.org/10.1038/s43247-023-01001-2.

Warrick, J.A., Vos, K., Buscombe, D., Ritchie, A.C., Curtis, J.A., 2023. A large sediment accretion wave along a Northern California littoral cell. J. Geophys. Res. Earth Surf. 128 (7), e2023JF007135. http://dx.doi.org/10.1029/2023JF007135.

Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., Bouwman, J., Brookes, A.J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C.T., Finkers, R., Gonzalez-Beltran, A., Gray, A.J., Groth, P., Goble, C., Grethe, J.S., Heringa, J., 't Hoen, P.A., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S.J., Martone, M.E., Mons, A., Packer, A.L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M.A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., Mons, B., 2016. The FAIR guiding principles for scientific data management and stewardship. Sci. Data 3 (1), 160018. http://dx.doi.org/10.1038/sdata.2016.18.

Wong, P.P., Losada, I.J., Gattuso, J.P., Hinkel, J., Khattabi, A., McInnes, K.L., Saito, Y., Sallenger, A., et al., 2014. Coastal systems and low-lying areas. Clim. Chang. 2104, 361–409.

Wulder, M.A., Roy, D.P., Radeloff, V.C., Loveland, T.R., Anderson, M.C., Johnson, D.M., Healey, S., Zhu, Z., Scambos, T.A., Pahlevan, N., Hansen, M., Gorelick, N., Crawford, C.J., Masek, J.G., Hermosilla, T., White, J.C., Belward, A.S., Schaaf, C., Woodcock, C.E., Huntington, J.L., Lymburner, L., Hostert, P., Gao, F., Lyapustin, A., Pekel, J.F., Strobl, P., Cook, B.D., 2022. Fifty years of landsat science and impacts. Remote Sens. Environ. 280, 113195. http://dx.doi.org/10.1016/j.rse.2022.113195.

Zhu, Y., Yuan, H., Wang, S., Liu, J., Liu, W., Deng, C., Chen, H., Dou, Z., Wen, J.R., 2024. Large language models for information retrieval: A survey. http://dx.doi.org/10.48550/arXiv.2308.07107, arXiv:2308.07107.