DEVELOPMENT OF AUTOMATED SATELLITE DATA DOWNLOADING AND PROCESSING PIPELINE ON AWS CLOUD FOR NEAR-REAL-TIME AGRICULTURE APPLICATIONS

A. Pandit[1]*, S.A. Sawant[1], R. Agrawal[1], J.D. Mohite[1], S. Pappula[2]

[1]TCS Research and Innovation, Tata Consultancy Services, Mumbai, India
[2]TCS Research and Innovation, Tata Consultancy Services, Hyderabad, India
(ankur.pandit, suryakant.sawant, rishabh.agrawal3, jayant.mohite, srinivasu.p)@tcs.com

Commission IV

KEYWORDS: Amazon Web Service, data downloading and processing, Sentinel-2, cloud-free, agriculture.

ABSTRACT:

Remote sensing satellites allow users to acquire detailed information about the Earth's surface on a temporal basis. Widen time-series analysis at a large geographical scale involves a huge amount (in Terabytes) of satellite data downloading and processing operations. Such processes need good computational power, large storage, and sophisticated tools. Maintaining such infrastructure can cost heavily to the research/commercial enterprises. To overcome such issues, Amazon Web Service (AWS) offers a sophisticated cloud computing environment. We developed an in-house automated satellite data downloading and processing (ADDPro) pipeline on the AWS platform. The ADDPro pipeline employed Sentinel-2 satellite data to offer current and relative vegetation health information of the agriculture region on a temporal basis at the pan-India scale. Image compositing and multi-sensor data fusion technique have been incorporated into the ADDPro pipeline to produce cloud-free raster (GeoTIFF) outputs. ADDPro pipeline also facilitates lossless raster data compression, which reduces AWS data transfer costs between regions. Data compression also aids in reducing raster publishing time on GeoServer. Operationally, AWS allows users to download only the bands required to generate a certain index (e.g. NDVI) rather than the entire Sentinel-2 data package. The entire ADDPro pipeline is extremely cost-effective, efficient, and scalable.

1. INTRODUCTION

Remote sensing has significantly contributed by providing consistent multi-sensor and multi-temporal data of larger regions with a good frequency of revisits and spatial resolution. In the past few years, the volume of remote sensing imagery has increased significantly. Nowadays, there is a huge demand for generating insights from the time-series remote sensing data across a wide range of industries such as urban planning and development, agriculture, insurance, climate change, etc. Our team at Tata Consultancy Services (TCS) Limited's Digital Farming Initiative (DFI) group (TCS, 2020) has extensively employed remote sensing datasets for performing client-oriented research and providing solutions for various agricultural applications. The major application involves the generation of time-series spatially distributed current vegetation health (CVH) and relative vegetation health (RVH) maps. The CVH is derived from the normalized difference vegetation index (NDVI) (Myneni et al., 1995) of the current year (e.g. $NDVI_{2022}$) whereas RVH is generated by subtracting the mean of the previous three years' NDVI [e.g. $mean(NDVI_{2021,\ 2020,\ 2019})$] from the current year NDVI (e.g. $NDVI_{2022}$), respectively. The NDVI is derived from the red and near-infrared reflectance ratio [NDVI = (NIR-RED)/(NIR+RED), where NIR and RED are the amounts of near-infrared and red light, respectively]. The other applications involve soil moisture estimation, crop yield estimation, crop variety classification, etc. Most of the agricultural parameters can be derived from the indices such as NDVI, normalized difference water index (NDWI), soil-adjusted Vegetation Index (SAVI), and others. These indices can be generated using the band combination of satellite datasets obtained from Sentinel-2/Landsat. For index generation, band operations need to be performed manually in dedicated software platforms such as ArcGIS or QGIS. This is a conventional method of processing satellite datasets, and it is appropriate only for regions where there are very few satellite datasets available. For larger geography such as a state or a country, deriving several indices has been a challenging task. This is mainly because of the huge amount of satellite data that must be downloaded and processed. This process may take weeks and several human resources if a conventional approach is adopted. Eventually, a typical approach is unsuitable, particularly in commercial enterprises where clients demand faster and more precise results at the state or country scale.

The limitations in the way of processing satellite data can be addressed by developing an automated satellite data downloading and processing pipeline (Pandit et al., 2020). The objective of such a pipeline is to download satellite datasets (e.g. Sentinel-1/2, Landsat) as per the requirement (i.e. dates, geography, processing level, etc.) and perform operations such as generation of NDVI, normalized difference water index (NDWI), soil-adjusted vegetation index (SAVI), and other indices. These indices can then be used in sophisticated models to derive insightful information related to agricultural use cases. The development of such a pipeline for executing processes across a greater geographic area necessitates enough

*Corresponding author

computing infrastructure, including storage capacity and processing power. Managing such infrastructure is expensive for businesses, especially small and mid-sized. So, alternatively, rather than purchasing and setting-up expensive infrastructure for big geo-spatial data operations, the costs can be reduced by using the resources provided by cloud computing services. Nowadays, several cloud computing services are available such as Google Cloud, Amazon Web Services (AWS), Microsoft Azure, Oracle, IBM, and a few others. Cloud computing is the on-demand delivery of information technology resources over the Internet. It works on the pay-as-you-go model under which instead of buying, owning, and maintaining physical data centers and servers, users can access technology services, such as storage, computing power, and databases, on an as-needed basis from cloud providers. Cloud service offers data security/protection at the highest level which makes organizations comfortable in running the business. Furthermore, cloud computing allows users to be more flexible by facilitating them to access data from anywhere using web-enabled devices such as laptops, notebooks, etc. As per the usage, cloud service platforms permit the user to rent extra processing power without having to use million-dollar machines as servers. Nowadays, many organizations are exhibiting interest in moving their applications to cloud environments instead of spending a lot of money on the hardware, software, licensing, and renewal fees.

Nowadays, many cloud-based platforms (mentioned previously) are offering services to use terabytes (TBs) to petabytes (PBs) of time-series data acquired by the selected satellites. One such cloud computing platform for earth observation data processing is offered by the Google Earth Engine (GEE), which was launched in 2010 by Google and is a proprietary system. GEE is conceptualized and designed to store and process huge satellite imagery and geospatial datasets at a multi-petabyte-scale (Gorelick et al., 2017). On a global scale, such datasets have been widely used by scientists, researchers, and developers in various applications e.g. land use/land cover (Saah et al., 2019), forest mapping (B. Chen et al., 2017), crop yield estimation (Chen et al., 2019), flood (Uddin et al., 2019), drought monitoring (Rembold et al., 2019), natural hazard management (Quintero et al., 2019), etc. The GEE library has over 800 functions for handling huge geospatial data sets. The services provided by GEE are completely free-of-cost for academic, research, and non-profit purposes, which bring a broad and growing userbase to the GEE platform. It can also be used for evaluation in a commercial or operational environment, but it can't be utilized in sustained production without a commercial license. Similar to GEE, the Sentinel-hub is another platform developed by Sinergise (Sinergise, 2020). Sentinel-hub is another service-oriented satellite imagery infrastructure, taking care of downloading, managing archives, and processing petabytes of satellite imagery, and making them available to end-users via simple-to-integrate web services. Its focus is on Sentinel satellites, however, additional support for Landsat and Planet is also provided by this platform. This service is also free-of-cost for non-commercial use, however, for commercial purposes, it charges as per their pricing plans. Paid service is having more features as compared to the free-of-cost service. Apart from these two satellite data processing platforms, some other platforms such as Open Data Cube (ODC) (Killough, 2018), System for Earth Observation Data Access, Processing and Analysis for Land Monitoring (SEPAL) (FAO, 2020), OpenEO (Pebesma et al., 2017), JEODPP (Soille et al., 2018), and pipsCloud (Wang et al., 2018) are also available for the same purpose but use different storage systems, access interfaces and abstractions for satellite data sets.

In the present study, we have developed an automated data downloading and processing (ADDPro) pipeline, which has been used to deliver pan-India scale *taluka* (a subdivision of a district) level cloud-free CVH and RVH rasters along with the *taluka-wise* zonal statistics. The pan-India scale temporal outputs have been derived using the Sentinel-2 satellite data.

2. MATERIALS

The overall ADDPro pipeline was developed and deployed on the AWS cloud computing environment using Python scripting language. The processing pipeline involves Sentinel-2 satellite data. The detail about each component involved in the pipeline is given in this section.

2.1 AWS Cloud Infrastructure

AWS brings in continuous integration and continuous delivery/deployment framework to accelerate product development and release cycles. In ADDPro pipeline development, we have mainly used two components of AWS- 1. Amazon Simple Cloud Storage Service (S3) and 2. Amazon Elastic Compute Cloud (EC2) instance.

Amazon S3 is a secure, durable, and scalable object storage infrastructure that allows users to store their huge amount of standard files in a bucket. It is built to automatically provide a high level of scalability and elasticity. Amazon S3 only charges for what we are currently using and there is no minimum fee. It has three pricing components: storage (per GB per month), data transfer in or out (per GB per month), and requests (per n thousand requests per month). We have mainly employed Amazon S3 for storing GBs of raster outputs, which have been obtained temporally by satellite data processing. The raster outputs have been uploaded and stored on Amazon S3 in two different locations. The first location is primarily used to store *taluka-wise* CVH raster data generated by the ADDPro pipeline. Rasters have been fetched and hosted on the GeoServer (an open-source server for sharing geospatial data) (Iacovella 2017) for visualization purposes from this location. The second location has been used for keeping tile-level current year and previous year(s) raster backups for future usage if any. The availability of tile-level outputs in the second location eliminates the need to download and reprocess massive Sentinel-2 data, saving time and computing resources.

Amazon EC2 instance is for running applications on the AWS infrastructure. Instance types are different combinations of CPU, memory, storage, and networking capabilities that allow users to choose the best resource combination for particular applications. AWS offers more than 60 On-Demand EC2 instances (AWS EC2 pricing). We have employed ml.c5.4xlarge notebook instance having a volume size of 4TB, 16 virtual central processing units, and 32 GB memory. The per-hour running cost of this instance is $0.816.

Package Name	Version	Purpose
glob	0.6	Use to capture patterns and supports recursive wildcards
numpy	1.17.4	Package used for array computing with Python
rasterio	1.1.0	Used for reading and writing geospatial raster data.
geopandas	0.6.1	Used for geospatial data operations
zipfile36	0.1.3	Used for reading and writing zip files.
pathlib	1.0.1	It offers a set of classes to handle file system paths.
shutil	3.5.2	offers several high-level operations on files and collections of files
json	3.1.1	Used for reading .json file
logging	0.4.9.6	Used for standard error logging
pyshp	2.1.3	Used for reading and writing ESRI shapefiles
sentinelsat	0.13	Used to search, download and retrieve the metadata for Sentinel products.
datetime	2.8.0	Used to work with the date as well as time
rasterstats	0.16.0	Module is used for summarizing raster product based on vector geometries. Also, it is used for zonal statistics and interpolated point queries.
Pyproj	3.3.0	Used for cartographic projections

Table 1. List of major Python packages

2.2 Satellite Data/Product

The Sentinel-2 (A and B) missions were launched by the European Space Agency under the Copernicus program. This mission provides multi-spectral data with 13 bands in the visible, near-infrared, and short wave infrared part of the spectrum, with 10 to 60 m spatial resolution. Sentinel-2 covers global landmasses once every 12 days (one satellite). The Copernicus Open Access Hub (ESA, 2022) provides complete, free and open access to Sentinel-2 products. On AWS, Sentinel-2 (Level 1C and Level 2A) scenes and metadata are available through the Requester Pays S3 bucket. Here, recently acquired Sentinel data are added frequently, within a few hours after they are available on Copernicus OpenHub. The major advantage of using the AWS S3 service is the possibility to download just selected bands of interest, for example, Band 4 and Band 8 for NDVI estimation, instead of the entire Sentinel-2 product. This facility greatly reduces the amount of data to be downloaded for a particular application. So, this allows us to download only selected bands (i.e., about a few MB) rather than downloading full data of about 600-800 MB. Apart from Sentinel-2 data, Moderate Resolution Imaging Spectroradiometer (MODIS) (Van Leeuwen et al., 1999) Vegetation Index Products produced at 16-day intervals have been used for image fusion. The Global Food Security-support Analysis Data (GFSAD) (Gumma et al., 2017) has been employed to extract agricultural regions from the respective tiles belongs the particular geometry.

2.3 Scripting environment and list of packages used

The pipeline was developed in the Python scripting language version 3.6. The AWS Software Development Kit (SDK) for Python (Boto3) enables developers to use Python code to interact with AWS services. The list of packages used in the development of python-based ADDPro pipelines is mentioned in Table 1.

3. STUDY SITE

The pipeline has been employed and tested explicitly over the agricultural region of the pan-India geography. Figure 1 represents the Indian geography with states and their corresponding *talukas*.

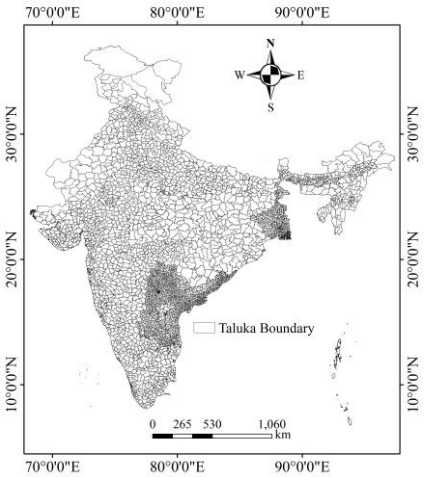

Figure 1. Map of India with *talukas*.

4. DEVELOPMENT OF AUTOMATED DATA DOWNLOADING AND PROCESSING PIPELINE

The top-level architecture of the ADDPro pipeline is shown in Figure 2. The overall AWS framework has three main components 1. Amazon Elastic Compute Cloud (EC2) instance, 2. Amazon Elastic Block Store (EBS), and 3. Amazon requester pay Simple Storage Service (S3) bucket. Amazon EC2 is a virtual server for running applications on the AWS infrastructure. EBS is a block-level storage volume attached to the EC2 instances. The collection of Sentinel-2 and other satellite imagery is located on the Amazon S3 bucket.

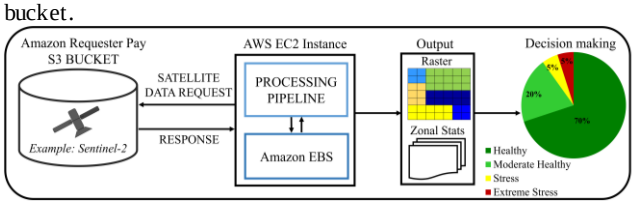

Figure 2. Top-level architecture of ADDPro pipeline

4.1 ADDPro pipeline for the regional-scale use case

The detailed flow chart of the ADDPro pipeline for generating CVH and RVH outputs at the pan-India scale is shown in Figure 3. It is important to highlight that we have estimated the pan-India scale NDVI index only, other indices (such as NDWI, SAVI, and others) can be calculated as per the requirement. The following steps are involved in the ADDPro pipeline-

1. To begin the procedure, the pipeline application must be supplied with the needed parameters listed in Table 2. The first five parameters in Table 2 are required for searching available Sentinel-2 data products from the Copernicus Open Access Hub for a specific geometry (n^{th} geometry resides in a given shapefile). Depending upon the shape and size of the geometry, the date range, and other search parameters, two scenarios may arise- (a) multiple Sentinel-2 products can be available for the particular geometry or (b) a single product can accommodate multiple geometries residing in a given shapefile. For the illustration purpose, the arrangement of geometry and its corresponding tiles is shown in Figure 4.
2. Once the list of products (1 to m) for n^{th} geometry is available, a distinct string was constructed for each product (e.g. m^{th} product), which was used to send a download request to the Amazon S3 bucket for the selected band. The bands have been downloaded according to the index that needs to be calculated. For example- Band 4 and Band 8 have been downloaded for NDVI calculation. Similarly, respective bands can be downloaded for other indices such as NDWI, SAVI, etc, if they need to be calculated. For all the indices, a cloud probability band has also been downloaded along with the respective bands for masking cloudy regions. In case, if no data product is available in the list, the application moves to the next geometry available in the shapefile and repeat step 1 and 2.
3. Then, based on the request, bands of m^{th} data have been downloaded from the Amazon S3 bucket to Amazon EBS attached with the instance, if not already available. It is also possible that the index from the product(s) belonging to a particular geometry has previously been generated. In that case, if the corresponding product is already processed and available in Amazon EBS, then the application directly subset the geometry region from that generated NDVI tile (Step 8). If the product is not processed, the process will move to Step 4.

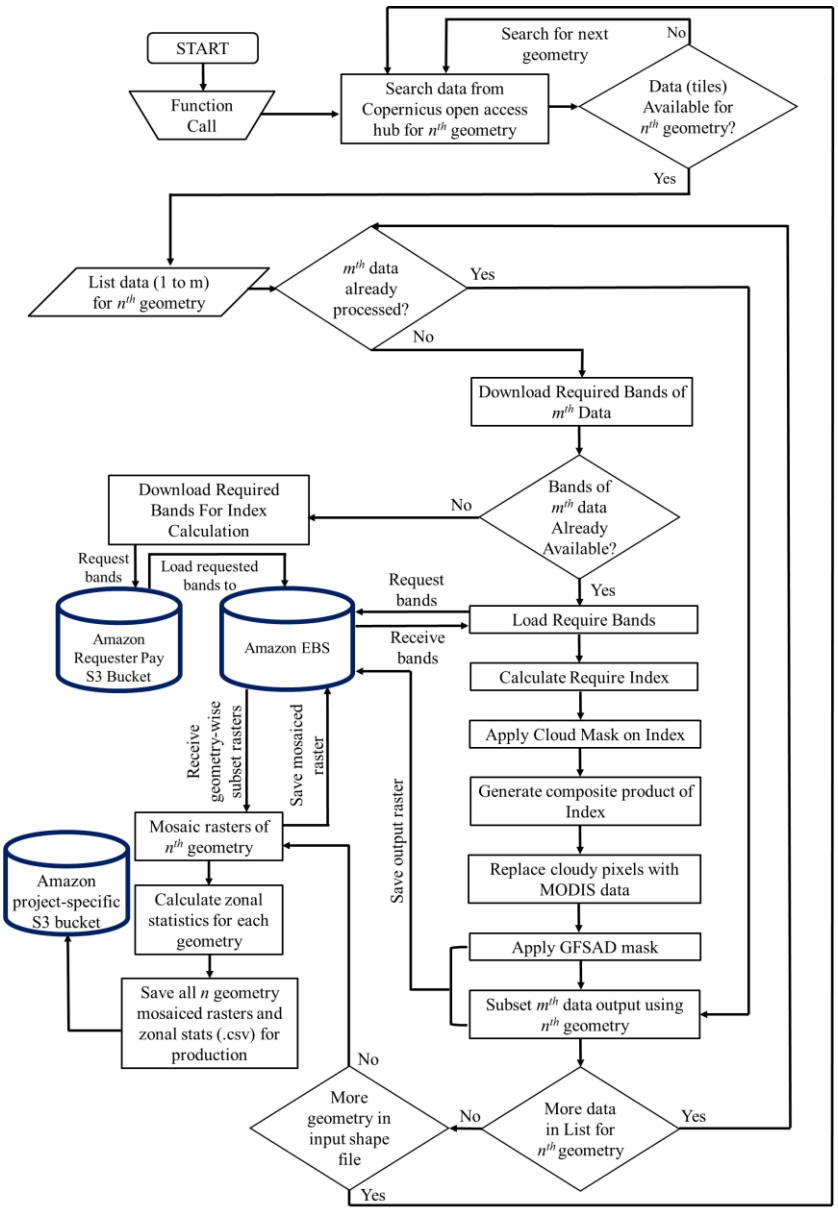

Figure 3. Detailed process flow diagram of ADDPro pipeline

Parameter / {Description}	Sample Input
S2_PROCESSING_LEVEL / {Level-1C- Top of the atmosphere, Level-2A- Bottom of the atmosphere}	[Level-2A/1C]
S2_FROM_DATE (Format: MMDD) / {Provide process start date}	'1201'
S2_TO_DATE (Format: MMDD) / {Provide process end date}	'1210'
S2_CLOUD_MAX_PER / {Provide maximum percentage of cloud cover allowed in Sentinel-2 data}	100
SHP_PATH / {Provide path of a shapefile consist of different geometries}	['./PAN_India/India.shp']
S2_LIST_INDEX / {Provide list of indexes need to generate}	['NDVI']
SESSION_ID / {Provide ID for particular crop season}	'RABI_2021'
S2_YEAR_TO_PROCESS_LIST (Format: YYYY) / {Provide list of years to process}	['2021', '2020', '2019', '2018']
S2_REQUIRED_PROC / {1= Download and process data 0= Only download the data}	1
RVH_YEARS_LIST (Format: YYYY) / {Provide list of years for RVH calculation}	['2021', '2020', '2019', '2018']
S2_ROOT_PROCESS_DIRECTORY / {Provide path of root directory}	'./Sentinel-2'
MODIS_TILE_LIST / {Provide list of tiles covering particular geography. Here, mentioned list is for Indian geography}	['h23v05', 'h24v05', 'h24v06', 'h25v06', 'h26v06', 'h24v07', 'h25v07', 'h26v07', 'h25v08']
MODIS_PROD_ID / {Provide ID of MODIS product}	['MOD13Q1.061']
S3_PREPROD_DIRECTORY / {Provide directory name of AWS S3 bucket for production purpose}	'pilot.sky.preprod'
S3_BACKUP_DIRECTORY / {Provide directory name of AWS S3 bucket for data backup purpose}	'pilot.sky.backup'
CLOSE_INSTANCE_AFTER_PROC / {0 means do not close the instance and 1 means close the instance after processing}	[1]
CREATE_MOSAICED_PRODUCT / {0 means perform mosaicing operation and 1 means do not perform mosaicing operation}	[1]

Table 2: List of parameters required for process initiation

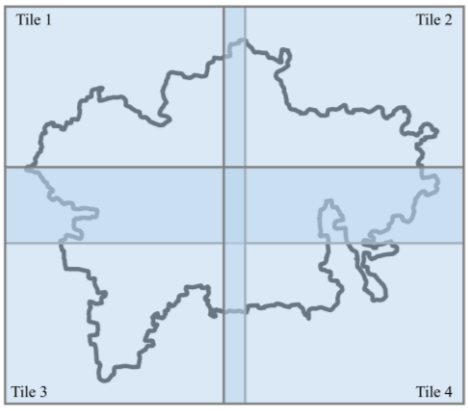

Figure 4. Geometry and its corresponding Sentinel-2 tiles

4. Next, the required bands have been loaded by the application for the generation of index (i.e. NDVI) raster. The NDVI raster has been named as *_NDVI.tif, where * represents [MissionIdentifer_TileId_YYYYMMDD] (for example- S2A_T46RBQ_20211222_NDVI.tif). Similar nomenclature has been followed at the successive stages of the pipeline for a particular index. Importantly, generated raster outputs (*_NDVI.tif) have been scaled to a range of 0 to 100 (from 0 to 1), resulting in small size output files (data format: int8) that are easy to publish and visualize on the GeoServer. Such lossless raster data compression also reduces AWS data transfer costs between regions.
5. A cloud probability band resampled to 10 m was then applied to the NDVI tile (*_NDVI.tif) to produce an NDVI raster tile without cloudy pixels (*_NDVI_CLDMSK.tif). Cloudy regions have been replaced with the NoData value in this case.
6. Further, the cloudy region (or pixels) in *_NDVI_CLDMSK.tif raster have been replaced by Sentinel-2 composite NDVI tile pixels, resulting in a composite raster output (i.e. *_NDVI_CLDMSK_COMP.tif) with very few cloudy pixels as compared to the prior raster (i.e. *_NDVI_CLDMSK.tif).
7. Next, the remaining cloudy pixels have been replaced with the 10 m resampled composite MODIS NDVI product (LPDAAC 2022). The generated product has been named as *_NDVI_CLDMSK_COMP_MODIS.tif. Later on, the only agricultural area has been extracted from the tile (*_NDVI_CLDMSK_COMP_MODIS.tif) using the GFSAD mask. So the final tile is the GFSAD masked raster i.e. *_NDVI_CLDMSK_COMP_MODIS_GFSAD.tif.
8. After the particular tile has been completely processed, the geometric region (full or partial) has been extracted from that tile and saved in the respective directory (./PATH/[Geometry_ID]/*_NDVI_CLDMSK_COMP_MODIS_GFSAD_SUBSET.tif).
9. It has been already discussed that particular geometry can be covered in multiple tiles (as depicted in Figure 4), therefore, the process will repeat Steps 2 to 8 for the next Sentinel-2 data (i.e. tile) belonging to that specific geometry. Here, all the tiles have been processed and saved inside the centralized repository, whereas, a subsetted part of geometry from the corresponding tiles has been saved in the separate directory that belongs to that particular geometry.

10. Once all geometries have been processed (that is, regions within the geometry have been retrieved from various tiles), the mosaicing operation has begun, in which subsetted tiles of each geometry have been mosaiced to generate a complete raster for that geometry.
11. After that, zonal statistics for all geometries (i.e. *taluka*) were generated and exported in a comma-separated values (.csv) file.
12. All rasters along with the zonal statistics have been migrated from Amazon EBS to the project-specific S3 bucket and then rasters have been published to the GeoServer. The output rasters and zonal statistics have been used for monitoring the vegetation growth/health, business, and policy decisions.

At the pan-India scale, each iteration of the ADDPro pipeline has been executed for ten days (e.g. 01-10 January 2022; 11-20 January 2022; 21-31 January 2022) so that the entire Indian geography can be covered. However, depending on the size of the target geography, the execution period (i.e. start and end date) may vary, for example- small geographies such as Goa and Kerala (states of India) can be covered in five days. Each iteration run generates a distinct log file that records information on each step (steps 1 to 12) of the processing pipeline. Logging aids in application troubleshooting and gives us insights into how our apps are doing on each of the many processing components.

5. RESULTS

Figure 5 represents the various intermediate rasters generated by the ADDPro pipeline for each tile that belongs to a specific geometry. Here, the initial tile-level output represents raw NDVI (i.e. with clouds), whereas the final tile-level output denotes an 8-bit compressed cloud-free NDVI raster with a GFSAD mask. Similarly, cloud-free NDVI rasters have also been generated for various tiles belonging to the same or different geometries.

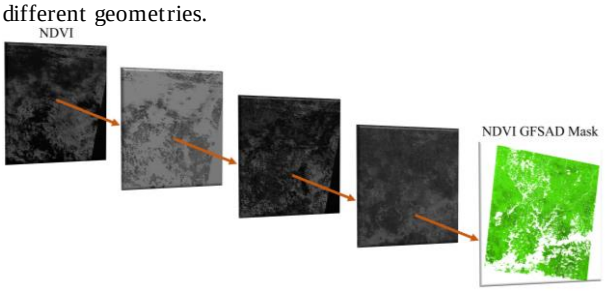

Figure 5. Illustration of raster tiles generated in multiple steps of ADDPro pipeline. NDVI with GFSAD mask is the final tile-level output

One of the generated cloud-free NDVI tiles encompassing several geometries of the given input shapefile is shown in Figure 6. Other tiles can also cover numerous geometries of the input shapefile depending on the shape and size of the geometry. Figure 7 represents the mosaiced raster of a particular geometry (i.e. *taluka*) generated by combining subsetted rasters extracted from the multiple processed tiles belonging to that particular geometry. Likewise, raster output has been generated for other geometries.

At the pan-India scale, the ADDPro pipeline has been executed three times in a month, every ten days (e.g. from 01-10 December, 11-20, and 21-30 December 2021). Figure 8 represents the pan-India scale spatial distribution of cloud-free CVH produced by publishing all *taluka*-level rasters together on GeoServer. Similarly, Figure 9 represents RVH for the same iterations.

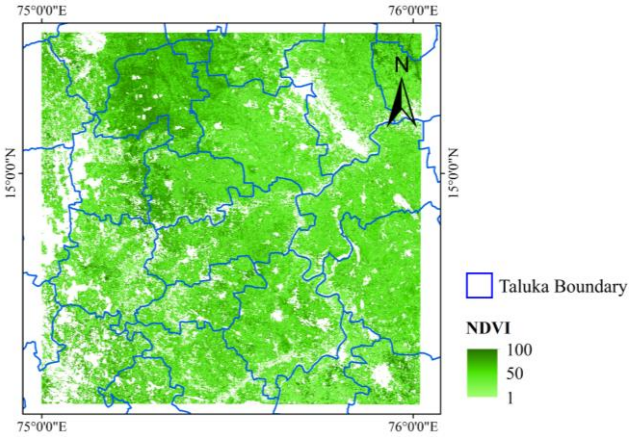

Figure 6. Processed cloud-free NDVI tile covering some of the geometries (i.e. *taluka*) of given input shapefile for 21-31 December 2021 iteration

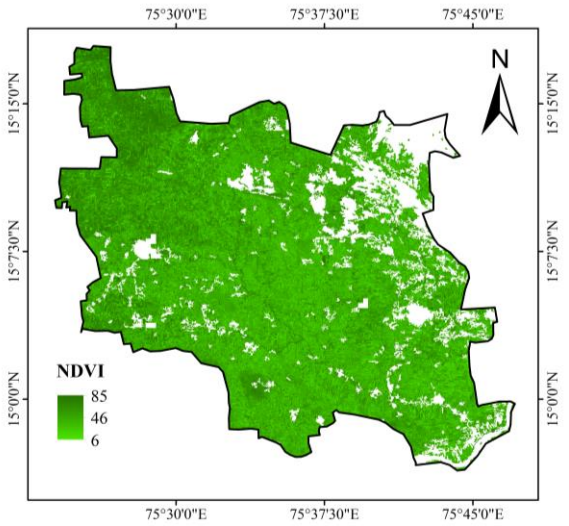

Figure 7. Mosaiced raster for a specific geometry (i.e. *taluka*) of given input shapefile for 21-31 December 2021

A useful collection of graphs/data visualization can be constructed based on the spatial distribution of CVH and RVH, which can aid in essential decision-making. For example- For five distinct *talukas*, Figure 10 quickly provides knowledge on how much acreage falls inside each of the NDVI ranges between two given dates (e.g. 11-20 December 2021). Similar information can also be obtained for other *talukas* and/or districts as per the requirement. At the pan-India scale, the overall processing takes around seventeen hours for a single iteration (e.g. 11-20 December 2021) to complete major operations of the complete data processing pipeline such as data downloading, index generation, and subsetting as per geometry, mosaicing, zonal statistics calculation, and uploading data to a predefined location on AWS S3 bucket.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-4-2022-189-2022 | © Author(s) 2022. CC BY 4.0 License.

6. DISCUSSIONS

At present, the ADDPro pipeline is commercially operational on the AWS platform for monitoring CVH and RVH during the Kharif as well as Rabi seasons across India. The problem of cloud cover especially during Kharif season is also managed by incorporating image compositing and multi-sensor data fusion technique in the ADDPro pipeline. For some of the most essential use-cases in the agriculture area, this in-house product eliminates the dependency on third-party applications such as GEE or Sentinel hub. The ADDPro pipeline can efficiently handle TBs of satellite data and provides a systematic way of managing enormous geospatial raster outputs. Thanks to the AWS cloud platform, which provides all of the resources required to develop and deploy the ADDPro pipeline. Handling big geospatial data is a critical element of the ADDPro pipeline in the current context when geospatial data is rising by the day (in TBs). Geometry-level rasters derived through the ADDPro pipeline can be used in a variety of applications and machine learning models that require the volume of time-series data. Segregation of large geographical data is accomplished efficiently in the overall ADDPro pipeline architecture so that users can extract data at any stage and use it meaningfully. Importantly, in this study, we have demonstrated how the ADDPro pipeline can perform operations over the larger Indian geography (about 3.2 million km^2). A country with a smaller geographical area than India can be undoubtedly monitored for temporal vegetation health. Even for a very large geographical area, time of processing and storage will be the only constraint.

7. CONCLUSION

For monitoring current and relative vegetation health on a temporal basis across pan-India, the ADDPro pipeline has been successfully implemented with the AWS cloud computing environment. The raster outputs have been made available for display and interpretation via the GeoServer. Furthermore, the zonal statistics created using output rasters are extremely useful for monitoring changes in vegetation distribution, productivity, and dynamics at the *taluka*-level. The cost of maintaining local infrastructure is eliminated by migrating the entire ADDPro pipeline to AWS, which is one of the major characteristics of cloud infrastructure. This pipeline is easy-to-scale-up and easy-to-deploy for any geographies across the globe, however, it requires standardization of input data. Using the ADDPro pipeline, TBs of satellite data for any given geography can now be downloaded and processed efficiently and effectively. Furthermore, once the pipeline is started, it does not require any manual user intervention. The pipeline can run 24 hours a day, seven days a week, during the day and at night, making temporal data processing for bigger geographies possible in a reasonable length of time.

ACKNOWLEDGEMENT

The authors are thankful to the Copernicus Open Access Hub for providing the list of Sentinel-2 datasets available during each iteration at the pan-India scale. The MODIS NDVI composite data was obtained on a temporal basis from the https://lpdaac.usgs.gov/.

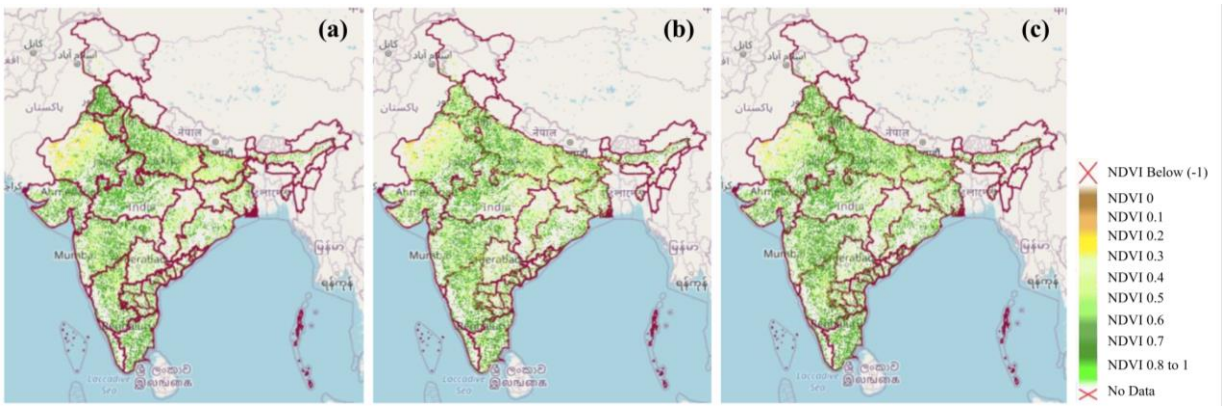

Figure 8. Pan-India scale spatial distribution of current vegetation health (CVH) produced by publishing *taluka*-level rasters on GeoServer a. 01-10, b. 11-20, c. 21-31 December 2021

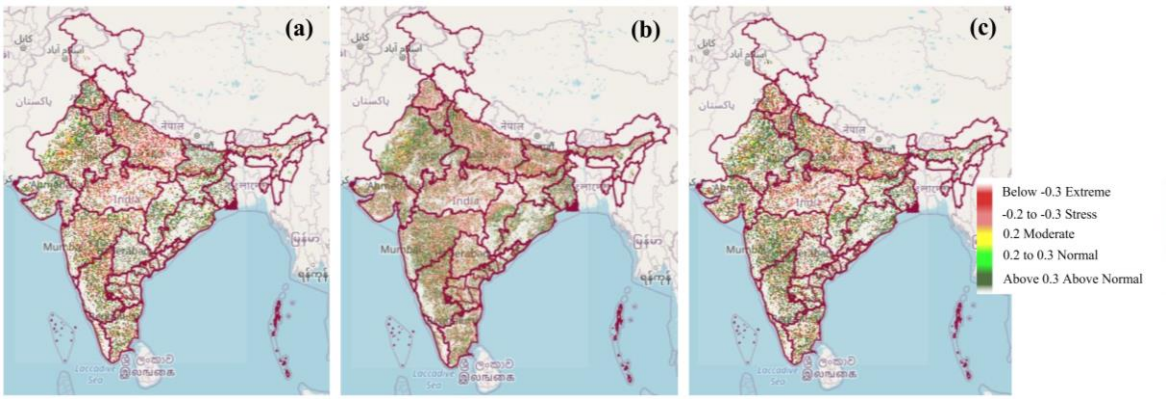

Figure 9. Pan-India scale spatial distribution of relative vegetation health (RVH) produced by publishing *taluka*-level rasters on GeoServer a. 01-10, b. 11-20, c. 21-31 December 2021

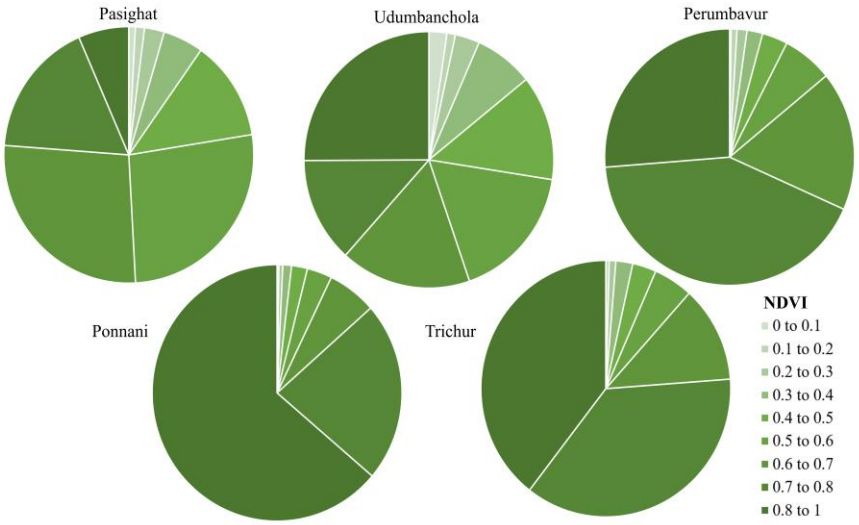

Figure 10. Illustration of *taluka*-wise percentage area falls under the different NDVI ranges. Graphs are generated from the zonal statistics (.csv) file obtained for the 11-20 December 2021 iteration

REFERENCES

Chen, B., et al., 2017. A mangrove forest map of China in 2015: Analysis of time series Landsat 7/8 and Sentinel-1A imagery in Google Earth Engine cloud computing platform. *ISPRS Journal of Photogrammetry and Remote Sensing*, 131, 104-120.

ESA, 2022. Retrieved from https://scihub.copernicus.eu/dhus/

FAO. SEPAL, a big-data platform for forest and land monitoring. Retrieved from https://www.fao.org/3/cb2876en/cb2876en.pdf

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R., 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote sensing of Environment*, 202, 18-27.

Gumma, M. K., Thenkabail, P. S., Teluguntla, P., Oliphant, A., Xiong, J., Congalton, R. G., ... & Smith, C., 2017. NASA Making Earth System Data Records for Use in Research Environments (MEASURES) Global Food Security-Support Analysis Data (GFSAD) Cropland Extent 2015 South Asia, Afghanistan, Iran 30 m v001.

Iacovella, S., 2017. GeoServer Beginner's Guide: Share Geospatial Data Using Open Source Standards. Packt Publishing Ltd.

Killough, B., 2018. Overview of the open data cube initiative. In *IEEE International Geoscience and Remote Sensing Symposium*, 8629-8632.

LPDAAC 2022. Retrieved from https://lpdaac.usgs.gov/

Pandit, A., Sawant, S., Mohite, J., Pappula, S., 2020. Development of Geospatial Processing Frameworks for Sentinel-1,-2 Satellite Data. *IEEE International Geoscience and Remote Sensing Symposium*, 3123-3126.

Pebesma, E., et al., 2017. OpenEO- A common, open source interface between Earth observation data infrastructures and front-end applications. Zendono/CERN, European Commission.

Quintero, N., Viedma, O., Urbieta, I. R., Moreno, J. M., 2019. Assessing landscape fire hazard by multitemporal automatic classification of Landsat Time Series using the Google Earth Engine in West-Central Spain. *Forests*, 10(6), 518.

Rembold, F., Meroni, M., Urbano, F., et al., 2019. ASAP: A new global early warning system to detect anomaly hot spots of agricultural production for food security analysis. *Agricultural systems*, 168, 247-257.

Saah, D., Johnson, G., Ashmall, B., et al. (2019). Collect Earth: An online tool for systematic reference data collection in land cover and use applications. *Environmental Modelling & Software*, 118, 166-171.

Sinergise. Sentinel Hub by Sinergise. 2020. Retrieved from https://www.sentinel-hub.com/

Soille, P., Burger, A., De Marchi, D., Kempeneers, P., Rodriguez, D., Syrris, V., Vasilev, V. 2018. A versatile data-intensive computing platform for information retrieval from big geospatial data. *Future Generation Computer Systems*, 81, 30-40.

TCS 2020. Big Data in Agriculture - Harnessing the Power of Sky and Earth, *TCS Digital Farming Initiatives - Indo-Canadian Workshop*. Retrieved from https://digitalsupercluster.ca/wp-content/uploads/2020/11/TCS_DFI_IndoCanadianWorkshop_Nov2020_vFinal.pdf

Uddin, K., Matin, M. A., Meyer, F. J., 2019. Operational flood mapping using multi-temporal sentinel-1 SAR images: a case study from Bangladesh. *Remote Sensing* 11(13), 1581.

Wang, L., Ma, Y., Yan, J., Chang, V., Zomaya, A. Y., 2018. pipsCloud: High performance cloud computing for remote sensing big data management and processing. *Future Generation Computer Systems*, 78, 353-368.

Myneni, R. B., Hall, F. G., Sellers, P. J., Marshak, A. L., 1995. The interpretation of spectral vegetation indexes. IEEE Transactions on Geoscience and Remote Sensing, 33(2), 481-486.

Van Leeuwen, W. J., Huete, A. R., & Laing, T. W. (1999). MODIS vegetation index compositing approach: A prototype with AVHRR data. Remote Sensing of Environment, 69(3), 264-280.