Analysing Vegetation and wetland cover of a district using GEE Python API and displaying the analysis in a Flask web application

Dr. Deepthi Murthy
School of Electronics and Communication Engineering
Reva University
Bangalore, India
deepthimurthy@reva.edu.in
Golakoti G V P Sai Ram
School of Electronics and Communication Engineering
Reva University
Bangalore, India
sairam.g581@gmail.com
Ch Cherishnav Reddy
School of Electronics and Communication Engineering
Reva University
Bangalore, India
ccreddy2214@gmail.com
Kadali Satyendra
School of Electronics and Communication Engineering
Reva University
Bangalore, India
kchandu4802@gmail.com
Golla Harish
School of Electronics and Communication Engineering
Reva University
Bangalore, India
harishrajeswari06@gmail.com

Abstract—Geographical analysis is one of the main applications of remote sensing, such as analyzing the vegetation and wetland areas. In general, Vegetation means a group of the same or different plant species covering a vast area. Vegetation includes different types of plant spices and geographical characteristics. Vegetation is classified based on life forms, structure, spatial extent, and botanical differences. In most tropical rainforests strenuous occurrence of deforestation has been seen. At present about thirty-one percent surface of Earth is spread by forests. Wetlands are the areas that contain a huge amount of water. Wetlands include rivers, lakes, ponds, tanks, etc. Evaluating and locating water bodies such as lakes, ponds, rivers, etc. along with vegetation areas is done by remote sensing. To monitor water bodies and vegetation areas remotely, remote sensing is very helpful. The images which are obtained through remote sensing are analyzed to determine the vegetation area and wetland area. Google Earth Engine is user friendly and easily accessible platform for remote sensing data. Google Earth Engine is helpful to access satellite data. Google Earth Engine contains many multi-spectral imaging satellites data such as Landsat, Sentinel, etc. Using GEE Python API, satellite data is analyzed. Forecasting can be used to predict the subsequent analysis of a certain district's area. A web application can be used as the user interface to display the analyzed and forecasted results.

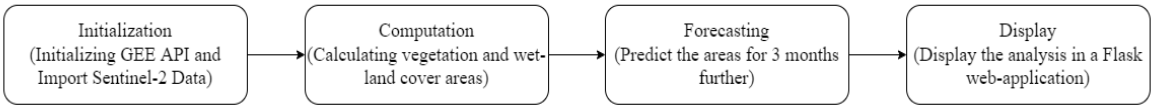
Keywords—remote sensing, google earth engine, vegetation cover, wetland cover

I. Introduction

Due to massive population growth and increased usage of land significantly led to environmental losses such as degradation of forests, the occurrence of soil erosion, and biodiversity losses. This phenomenon in the environment brought great pressure on human life. As we can see in our day-to-day life, the deforestation rate has rapidly increased. On the other hand, we can also see that the groundwater level is severely reduced. Due to increased mining activities, vegetation and surface water are extremely affected which in turn results in reduced vegetation area and wetlands. Remote sensing can be used to monitor and analyse these activities which can be used to prevent further loss. The analysis of vegetation cover and wetland cover can be made on a selected area of interest (AOI). An area of interest can be any geographical boundary like a city or a district. The district boundaries of India are considered as areas of interest in this project. Google Earth Engine (GEE) is a powerful tool that comprises remotely sensed data from various satellite imaging sensors such as Landsat, MODIS, and Sentinel. The GEE Python API is used in this project to analyse and forecast the data. Python is used to develop a web application to display the analysis and forecast results. Sentinel-2 is a multispectral imaging sensor that perceives the data of the earth in a wide range of spectrum bands and is available through GEE API. It consists of different bands to analyse vegetation and wetland cover which are the two main objectives of this project. A forecasting machine learning model built with Python is trained on the data to predict the analysis for further months. Flask module in Python is used to display the results of the analysis on a web application on a local server.

II. LITERATURE REVIEW

In 2007, An overview [1] was published by Ranganath R on applications of Remote sensing. His research article includes details about different sensors used for remote sensing with their host satellite. Spectral bands, spatial resolution, swath, and



other specifications of the sensors are also tabulated. The article also gives an overview of basic concepts like Signatures and remote sensing techniques.

Indersheel analysed the deforestation in Maharashtra [2], India using the satellite image obtained through remote sensing in 2021. He used the multispectral data for the analysis. The total area of study is 107665.5 hectares. After performing image processing and image enhancement methods, he analysed that the area of forest cover is 76.740.32 hectares. With the study, he concluded that there is a significant decrease in water bodies percentage and forest cover.

Ila Chawla, a Water Researcher at IISc inscribes a review on applications of remote sensing on water security in 2020 [3]. She summarizes the role of satellites to evaluate water security through remote sensing. She mainly focuses on the three characteristics of water that are quality, quantity, and extremes. To assess the quality of water, she considers the Chlorophyll-a, TSS, SDD, and CDOM parameters. She gives the outline of the application of remote sensing to identify streamflow, water storage bodies, and reservoir monitoring.

Figure 1 Block diagram

Yan Gao with the team reviewed the role of remote sensing in forest degradation in 2020 [4]. In the article, they listed the definition of forest degradation and the limitations of remote sensing data. They also listed the disturbances for forest degradation.

GEE has a wide range of applications as reviewed by Lalit Kumar [5]. He presents a summarized review of the applications of GEE across different categories. This includes categories like vegetation and landcover mapping, and agricultural and disaster management applications of GEE.

III. METHODOLOGY

A. Block diagram

The block diagram in figure 1 represents the methodology for the analysis. After initializing the API, required data is computed in Python, and the results are forecasted and displayed in the Flask application.

1) Initialization

This step starts with initializing the Google Earth Engine Python API and authenticating it with the credentials to get started. After initializing the API, The Sentinel-2 image collection is imported for the area of interest in this case certain district data.

```
import ee
ee.Authenticate()
ee.Initialize()
image_collection=ee.ImageCollection("COPERNICUS/S2_SR")
```

2) Computation

The imported image collection is grouped by the image’s captured year and month. Normalized differences for vegetation and water are calculated. The normalized difference images are converted into a gif file to illustrate. The areas of the vegetation and wetland are computed and stored in a data frame.

3) Forecasting

Vegetation and wetland cover areas are then forecasted using a Forecasting Machine Learning Model for further 3 months from the last data point. This step returns the forecast results and time-series graphs for better representation.

4) Display

A Flask-based web application is used as an interface to provide the inputs and display the results. The Flask application runs on a local server and is used to take input from the user and display the appropriate results.

B. Data

The data used for the analysis is Sentinel-2 Level-2A data. Sentinel-2 is a multi-spectral high-resolution instrument sampling in 13 different spectral bands ranging from 443nm to 2190nm labeled as B1 through B12 with 4 bands at 10m, 6 bands at 20m, and 3 bands at 60m spatial resolutions. It is used for Copernicus Land Monitoring. Sentinel-2 Level-2A is one of the products of Sentinel-2 data with Bottom-Of-Atmosphere reflectance and other classes for different clouds. The data is available from March 20018 to April 2022 (as of this article).

C. Flow Cart

Figure 2 shows the detailed flow graph. When accessed, the Flask application displays a form to take the district name from the user. Geo-JSON is a type of data that contains geographical information such as borders, landmarks, and other user-defined data which are identified by the coordinates. The border of the district is used as the feature in this project. The user-provided district's border feature is filtered from the Geo-JSON data and is converted to a GEE API supported feature object and the total area covered by the border is calculated. The Sentinel-2 data from GEE API is imported as an image collection and is filtered for the respective district with a cloud cover of 30% and a mean image for each month is computed. The algorithm used to identify the vegetation and water body is Normalized Difference. When computed, normalized difference returns values in the range of -1 to 1.

Normalized Difference Vegetation Index (NDVI):

NDVI is used to identify the vegetation cover. NDVI is a dimensionless value that represents the reflectance of vegetation cover. It is determined from the reflectance values of the vegetation cover in the red spectrum (RED) with a wavelength in the range of 664.5nm to 665nm and the Near-Infrared band (NIR) with a wavelength in the range of 833nm to 835.1nm [6]. NDVI is given as equation (1).

$$NDVI=(NIR-RED)/(NIR+RED) \quad (1)$$

For Sentinel-2 data, band B8 of 10m spatial resolution represents the NIR band with a central wavelength of 832.8nm and bandwidth of 106nm, and band B4 of 10m spatial resolution represents VIS in the red spectrum with a central wavelength of 664.6nm and bandwidth of 31nm. Any NDVI value when computed is less than or equal to zero, it represents the water cover. Values from 0.1 to 0.3 represent the presence of stone and snow and values greater than 0.3 represent vegetation. The larger the value, the denser the vegetation cover.

Normalized Difference Water Index (NDWI):

NDWI is used to identify the wetland cover. NDWI is a dimensionless value that represents the reflectance of wetland cover. It is determined from the reflectance values of the vegetation cover in the green spectrum with a wavelength in the range of 559nm to 560nm and the Near-Infrared band with a wavelength in the range of 833nm to 835.1nm [7]. NDWI is given as equation (2)

$$NDWI=(GREEN-NIR)/(GREEN+NIR) \quad (2)$$

For Sentinel-2 data, band B3 of 10m spatial resolution represents the green spectrum with a central wavelength of 559.8nm and bandwidth of 36nm, and band B8 of 10m spatial resolution represents the NIR band with a central wavelength of 832.8nm and bandwidth of 106nm. Values greater than zero represents water bodies.

The NDVI and NDWI algorithms are used on all the images in the imported Earth Engine Image Collection. After identifying the vegetation and wet-land cover, vegetation and wet-land cover areas are calculated using the Google Earth Engine Python API functions in square kilometers for all the images in the collection. Area values of each image are stored in a Pandas data frame with the date of the image as the index and a time-series graph is plotted for the display.

The data in the data frame is used to create a Machine Learning model to predict the vegetation and wet-land cover areas for further 3 months and plot the prediction graphs. The recursive Autoregressive Forecasting model [8] is chosen as the machine learning model for the forecast and Random Forest Regressor is passed as the regressor for the forecasting algorithm with 6 lags (fig 3).

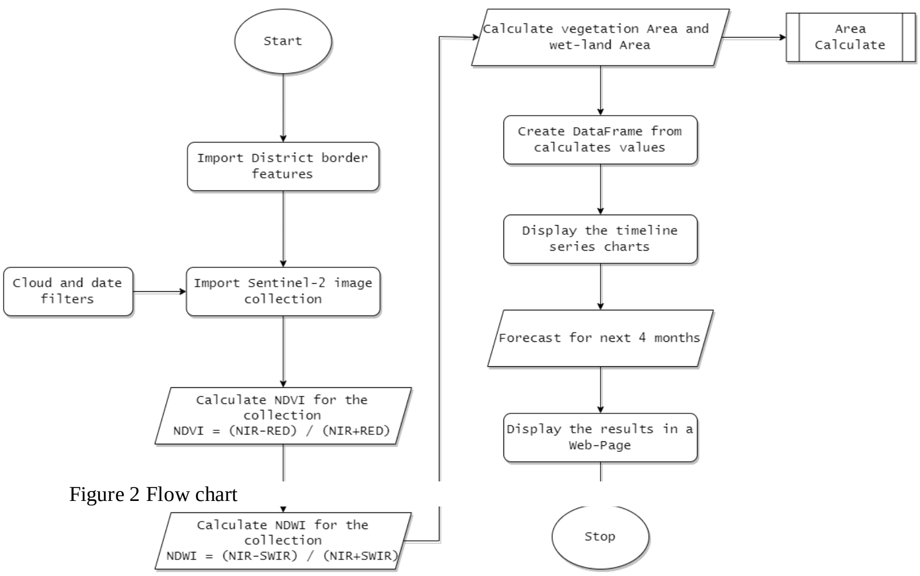


Figure 2 Flow chart

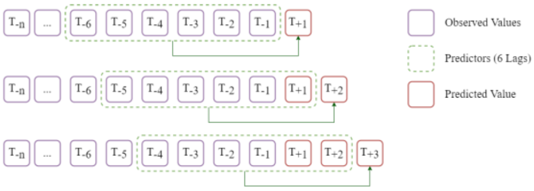


Figure 3 Recursive Multi-step Forecasting with 6 lags

All the resulted values and plots are then displayed on the display page of the Flask-based web application.

IV. RESULTS

The Flask web application is designed to make the user choose the district for the analysis and display the results on a separate page. The web application is developed in HTML and CSS for the home and result display page as the frontend and Flask is used as the backend for routing and computing required analysis. It runs on a local development server to provide the service. The Home page of the application consists of a simple form with a drop-down with the districts to select from and a button to submit the form as shown in figure 4.

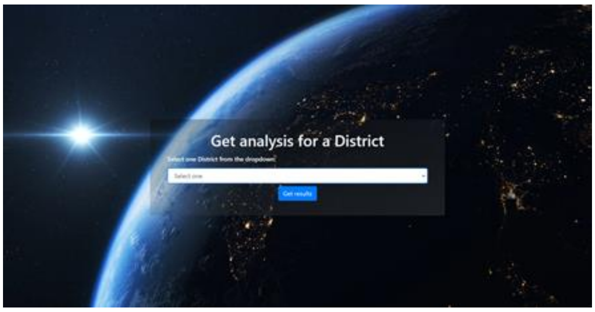


Figure 4 Application homepage with form to select district

The below screenshots are the results when East Godavari is selected as the district. When submitted, the application starts computing the results for the district and displays a loading animation till the output is generated as shown in figure 5.

Figure 5 Application loading status

After computing, the results display page shows the name of the selected district along with the total area of the district in square kilometers and a recent true-color image of the district (fig 6).

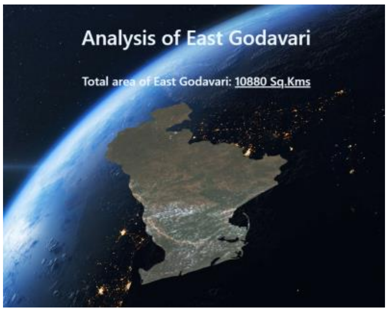


Figure 6 True color image of the district with total area (Sq.KM)

The changes in the vegetation cover and wetland cover are presented as gifs with the respective areas as shown in figure 7.

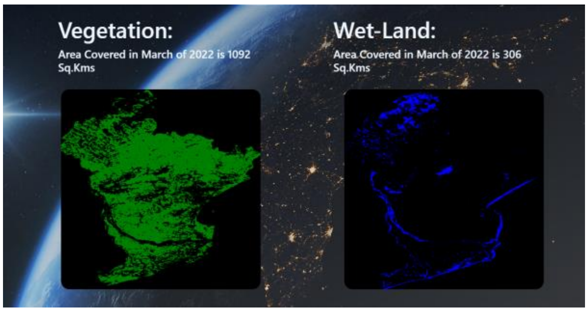


Figure 7 Result with recent vegetation and wetland cover areas and their representation

Figure 8 displays two plots representing time-series of vegetation (top) and wetland (bottom) covers with the y-axis representing the area in square kilometers and the x-axis representing the timeline.

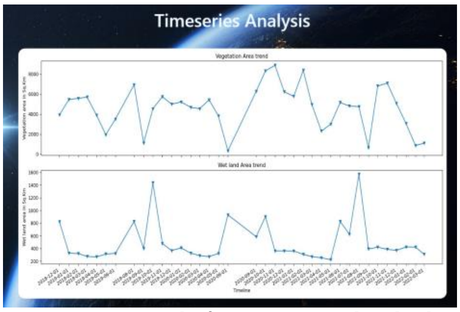


Figure 8 Timeseries graphs for vegetation and wetland covers

Figure 9 represents the forecasting analysis with the left plot representing the vegetation cover forecast and the right plot representing the wetland cover forecast. The train and test data are represented by blue and orange lines respectively and the green line represents the predicted values. Also, the created model is used to predict the areas for successive 3 months from the last data point.

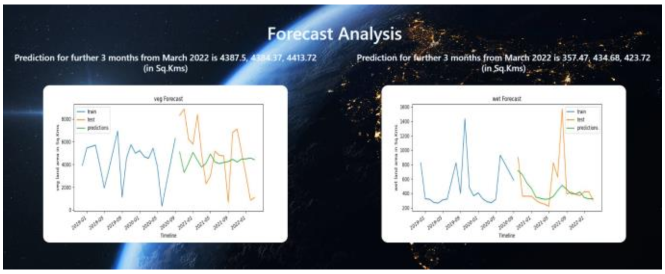


Figure 9 Forecasting for vegetation and wetland covers

We can observe that there are outliers present in the data from figure 9 like the values almost 0 which cannot occur in real-time. This is due to errors while filtering the image collection, we may obtain images with partial coverage of the AOI and cloud parameters.

V. FUTURE SCOPE

- This project can be carried on to analyze mineral contents using Python and GEE and display it on the webpage.
- Extending the area of interest from districts to other boundaries can be implemented.
- Outliers obtained by data filter errors like partial image and cloud cover handling will yield better results.

VI. CONCLUSION

This project is able to analyze the vegetation and wetland covers of a district with the help of Google Earth Engine API and predict the areas for the successive 3-month period with a Flask-based web application as the interface. Generally, forecasting is used to predict the weather and climatic conditions of an area. Forecasting can also be used to foresee the vegetation and wetland changes. Using a Flask-based web application as an interface helped the integration of the analyzing and forecasting code in Python with the webpage.

REFERENCES

[1] Navalgund, Ranganath R., V. Jayaraman, and P. S. Roy. “Remote Sensing Applications: An Overview.” Current Science 93, no. 12 (2007): 1747–66. http://www.jstor.org/stable/24102069.

[2] I. K. H. S. Nanare, D. B. Bhoyar and S. V. Balamwar, "Remote Sensing Satellite Image Analysis for Deforestation in Yavatmal District, Maharashtra, India," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), 2021, pp. 684-688, doi:10.1109/ICSPC51351.2021.9451744.

[3] Ila Chawla, L. Karthikeyan, Ashok K. Mishra, A review of remote sensing applications for water security: Quantity, quality, and extremes, Journal of Hydrology, Volume 585, 2020, 124826, ISSN 0022-1694, https://doi.org/10.1016/j.jhydrol.2020.124826.

[4] Yan Gao, Margaret Skutsch, Jaime Paneque-G lvez and Adrian Ghilardi, Remote sensing of forest degradation: a review, 2020, https://doi.org/10.1088/1748-9326/abaad7

[5] Mutanga, O.; Kumar, L. Google Earth Engine Applications. Remote Sens. 2019, 11, 591. https://doi.org/10.3390/rs11050591

[6] Schmid, Jan Niklas. (2017). Using Google Earth Engine for Landsat NDVI time series analysis to indicate the present status of forest stands. 10.13140/RG.2.2.34134.14402/6.

[7] S. K. McFEETERS (1996) The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features, International Journal of Remote Sensing, 17:7, 1425-1432, DOI: 10.1080/01431169608948714

[8] Bontempi, Gianluca & Ben Taieb, Souhaib & Le Borgne, Yann-Aël. (2013). Machine Learning Strategies for Time Series Forecasting. 10.1007/978-3-642-36318-4_3.