Hacettepe University Department of Geomatics Engineering

2024 – 2025 Spring

GMT447 Digital Image Processing

Homework #1

Sentinel-2 Image Stacking and Analysis in Google Earth Engine

Author: Gülüzar Yiğit 2200674020

19.04.2024

Objective

The goal of this assignment is to perform image stacking and statistical analysis of Sentinel-2 imagery using Google Earth Engine (GEE). The steps include importing data, filtering based on cloud percentage, resampling, creating stacked images, and calculating statistical metrics including mean, standard deviation, covariance, and correlation matrices.

Code and Steps

1. Importing Sentinel-2 Data and Defining Study Area

The Sentinel-2 Level-2A dataset was loaded, and the study area was defined using polygon geometry.

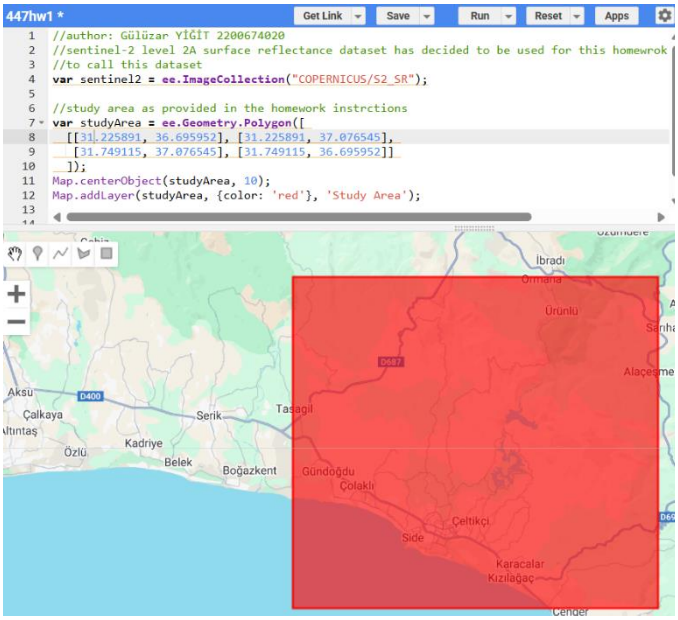
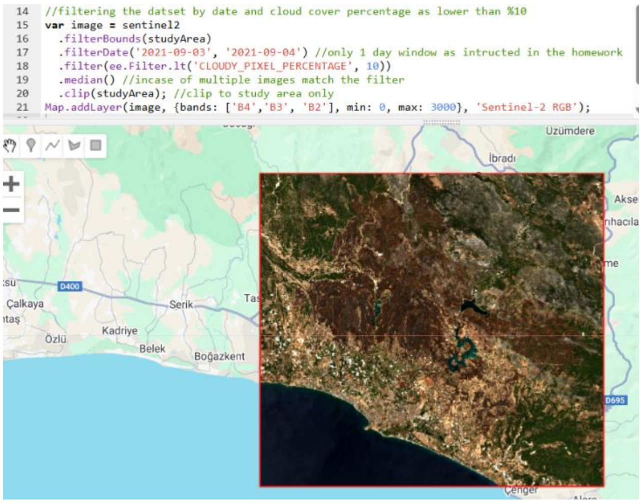


Figure 1 Definition of the study area in the Google Earth Engine environment.

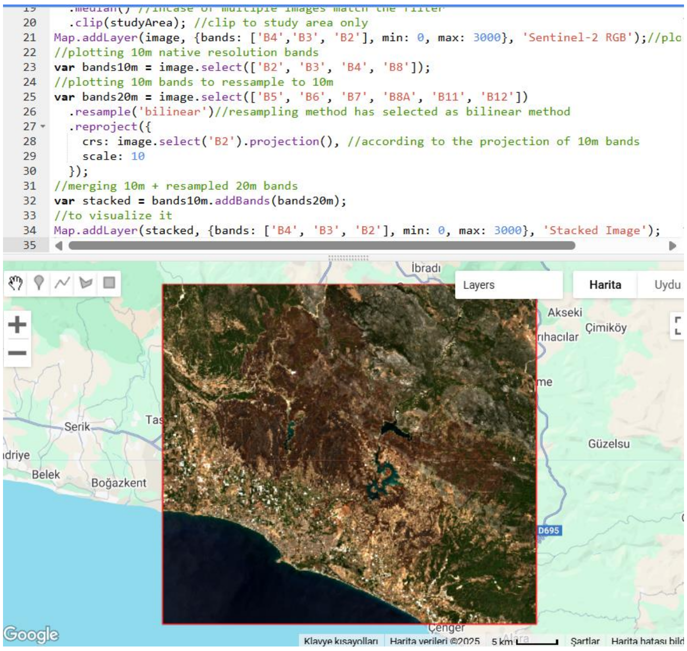
2. Filtering by Date and Cloud Cover



The image was filtered by date and limited to less than 10% cloud cover to avoid cloud-contaminated pixels. It was then clipped to the study area for spatial focus and visualized using RGB bands (B4, B3, B2).

Figure 2 Cloud filtering and RGB visualization.

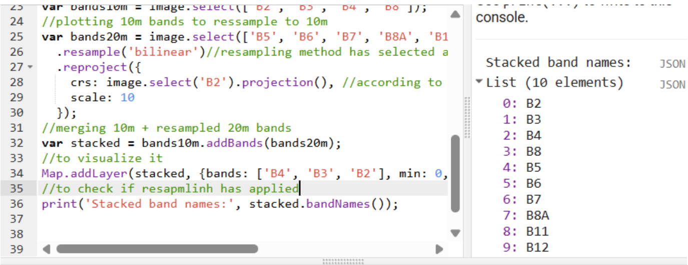
3. Resampling 20m Bands to 10m and Stacking



Since Sentinel-2 bands have different spatial resolutions, the 20 m bands were resampled to 10 m using bilinear interpolation to match the native resolution of the 10 m bands. After resampling with the reproject () method, all bands were stacked using addBands.

Figure 3 Resampled and stacked RGB image.

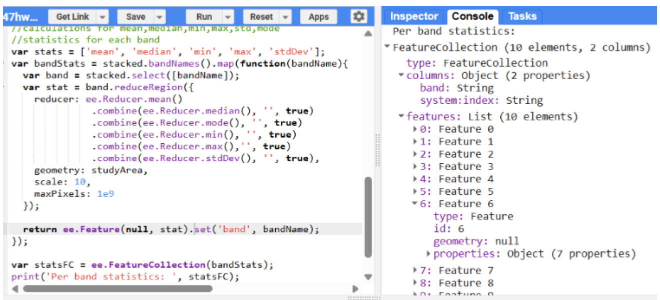
4. Per Band Statistics (Mean, Median, Mode, Min, Max, StdDev)



After resampling, all 10 m and resampled 20 m bands were merged into a single stacked image. The bandNames() function was used to verify that all bands were successfully included in the stacked image.

Figure 4:Band list check

5. Covariance and Correlation Matrices



The reduceRegion() function was applied over the study area to compute statistical metrics—mean, median, mode, min, max, and standard deviation—for each band. These statistics describe the distribution of pixel values per band and were returned as a FeatureCollection.

Figure 5:Statistical calculations (per-band)

```
});

var statsFC = ee.FeatureCollection(bandStats);
print('Per band statistics: ', statsFC);
//to calculate inter-covariance and inter-correlation
//firstly, converting stacked image to an array image
var arrayImage = stacked.toArray();
//compute covariance
var covarianceArray = arrayImage.reduceRegion({
  reducer: ee.Reducer.covariance(),
  geometry: studyArea,
  scale: 10,
  maxPixels: 1e9
});
print('Covariance matrix: ', covarianceArray);
print('Band order:', stacked.bandNames());
//compute correlation
//to convert the covariance matrix to eeArray
var covArray = ee.Array(covarianceArray.get('array'));
//standart deviations of the bands:
var stdList = stacked.reduceRegion({
  reducer: ee.Reducer.stdDev(),
  geometry: studyArea,
  scale: 10,
  maxPixels: 1e9
});
```

```
Covariance matrix:
▾Object (1 property)
  ▾array: List (10 elements)
    ▸0: List (10 elements)
    ▾1: List (10 elements)
      0: 284492.5793414435
      1: 337465.9815079153
      2: 402349.09941609774
      3: 372074.92038065166
      4: 1199.9732007068021
      5: 2174.5876090432134
      6: 2482.273045122006
      7: 2695.1581035979143
      8: 1982.4921070587202
      9: 1219.9311749391647
    ▸2: List (10 elements)
    ▸3: List (10 elements)
    ▸4: List (10 elements)
    ▸5: List (10 elements)
    ▸6: List (10 elements)
    ▸7: List (10 elements)
    ▸8: List (10 elements)
    ▸9: List (10 elements)
```

```
▾6: Feature 6
   type: Feature
   id: 6
   geometry: null
  ▾properties: Object (7 properties)
     B7_max: 1874.8687669153778
     B7_mean: 1283.328153534432
     B7_median: 1267.9165312662537
     B7_min: 737.9250107223494
     B7_mode: 747.9794066822894
     B7_stdDev: 328.0224091485409
     band: B7

▸7: Feature 7
▸8: Feature 8
▸9: Feature 9
```

Figure 7:Statistical calculations of Band 7 as example

Figure 6: Covariance matrix result

The covariance() reducer was used to compute the covariance between all band pairs within the study area (Figure 6). This helped quantify how changes in one band are associated with changes in another. As shown in Figure 7, per-band statistical metrics (mean, median, min, max, mode, stdDev) were calculated for Band 7 as an example.

```
Band order:
▾List (10 elements)
   0: B2
   1: B3
   2: B4
   3: B8
   4: B5
   5: B6
   6: B7
   7: B8A
   8: B11
   9: B12
```

Figure 8: Band order list

$$\mathrm{corr}(X, Y) = \frac{\mathrm{cov}(X, Y)}{\sigma_X \cdot \sigma_Y}$$

Figure 9: Correlation formula

Since Earth Engine does not have a built-in correlation reducer, the correlation matrix was calculated manually. First, the covariance matrix was converted into an ee.Array. Then, standard deviations were computed and outer-multiplied to obtain the denominator. Finally, element-wise division (divide()) yielded the 10×10 correlation matrix. Each row in the output corresponds to a specific band index (axis 0), while i, i+1 slicing isolates the selected row.

```
print('Covariance matrix: ', covarianceArray);
print('Band order:', stacked.bandNames());
//compute correlation
//to convert the covariance matrix to eeArray
var covArray = ee.Array(covarianceArray.get('array'));
//standart deviations of the bands:
var stdList = stacked.reduceRegion({
  reducer: ee.Reducer.stdDev(),
  geometry: studyArea,
  scale: 10,
  maxPixels: 1e9
});
var stdNames = stacked.bandNames(); // ['B2', ..., 'B12']
var stdValues = stdNames.map(function(name) {
  return stdList.get(name);
});
var stdArray = ee.Array(stdValues).reshape([10, 1]);
//by outer product, create std. dev.'s transpose
var denom = stdArray.matrixMultiply(stdArray.transpose());
var corrArray = covArray.divide(denom);
print('Corrected Correlation Matrix:', corrArray);
//to see the correlations inter-band
for (var i = 0; i < 10; i++) {
  var row = corrArray.slice(0, i, i + 1);
  print('Correlation row for band index ' + i + ':', row);
}
```

```
Corrected Correlation Mat
▾List (10 elements)
  ▸0: List (10 elements)
  ▾1: List (10 elements)
     0: 0.8606064179925248
     1: 0.8603135126506778
     2: 0.8152208006413828
     3: 0.5151025440462154
     4: 0.012053935025661355
     5: 0.012053935025691942
     6: 0.01205393502583197
     7: 0.012053935025892882
     8: 0.012053935025799984
     9: 0.012053935025764595
  ▸2: List (10 elements)
  ▸3: List (10 elements)
  ▸4: List (10 elements)
  ▸5: List (10 elements)
  ▸6: List (10 elements)
  ▸7: List (10 elements)
  ▸8: List (10 elements)
  ▸9: List (10 elements)
```

The matrix shows all pairwise correlation coefficients between bands, such as B2–B2, B2–B3, ..., B12–B12. Here, axis = 0 refers to rows (band rows), and i, i+1 is used to slice and display each row individually.

Figure 10:Correlation matrix code and result

```
Correlation row for band index 0:
▾List (1 element)
  ▾0: List (10 elements)
     0: 0.9331561442570844
     1: 0.8606139947145512
     2: 0.8034556648345846
     3: 0.4235425639222217
     4: 0.003078379043292785
     5: 0.0030783790433017814
     6: 0.0030783790433374213
     7: 0.0030783790433517263
     8: 0.0030783790433289047
     9: 0.0030783790433200364
```

Figure 11: Correlation values for band

```
Correlation row for band index 1:
▾List (1 element)
  ▾0: List (10 elements)
     0: 0.8606139947145512
     1: 0.8603354907905525
     2: 0.815246346213266
     3: 0.5151373457303485
     4: 0.012085158626616983
     5: 0.012085158626647656
     6: 0.012085158626788103
     7: 0.012085158626849184
     8: 0.012085158626756033
     9: 0.012085158626720547
```

Figure 12:Correlation values for band

6. Exported Image

The final stacked image was exported as a GeoTIFF file using the Export.image.toDrive() function in Google Earth Engine. The export was performed in the EPSG:4326 coordinate reference system, at a spatial resolution of 10 meters. The exported file is named GMT447_Stacked_Manavgat.tif, and is accessible via the following Google Drive link:
https://drive.google.com/file/d/1kDQ4j7-QuSY3B33TES4DSsYbIG6yAAe9/view?usp=drive_link

Conclusion

In this assignment, Sentinel-2 Level-2A surface reflectance imagery was successfully utilized to create a stacked multi-band image covering the designated study area near Manavgat. The workflow involved importing and filtering the image collection based on spatial bounds, acquisition date, and a cloud cover threshold. Bands originally provided at 20m spatial resolution were resampled to 10m using bilinear interpolation, then merged with 10m native resolution bands to produce a uniform-resolution stacked image.

Following the construction of the stacked image, several statistical analyses were conducted to assess per-band distributional properties (mean, median, standard deviation, etc.) and inter-band relationships. The covariance matrix was calculated to quantify the degree of joint variability between the spectral bands. To standardize and interpret these relationships more effectively, the covariance matrix was converted into a correlation matrix.

The correlation results demonstrated strong linear associations among certain bands. For instance, Band 2 (Blue) showed a very strong correlation with Band 3 (Green) ($r \approx 0.86$) and Band 4 (Red) ($r \approx 0.80$), which aligns with expectations given the overlapping spectral ranges and similar reflectance behavior in visible wavelengths. Meanwhile, its correlation with Band 8 (NIR) was moderate ($r \approx 0.42$), indicating differing reflectance properties of near-infrared data compared to visible bands.

Furthermore, Band 3 (Green) exhibited a very high correlation with Band 4 (Red) ($r \approx 0.82$) and also a relatively high correlation with Band 8 ($r \approx 0.51$). These observations are consistent with typical vegetation reflectance patterns, where the transition between visible and NIR bands results in varied spectral responses.

Conversely, both B2 and B3 showed negligible correlation with SWIR bands (B11, B12) and red-edge bands (B5–B7, B8A), with correlation values near zero (e.g., $r \approx 0.003$–0.012). This suggests that these spectral regions provide complementary information and are largely uncorrelated with the visible spectrum, which is beneficial for applications like land cover classification or vegetation stress detection.

In summary, the methods employed in this study enabled the preparation of a spatially consistent stacked image and the exploration of inter-band spectral relationships. The correlation matrix particularly highlighted redundancy within visible bands and independence across different spectral regions. These findings reinforce the importance of spectral diversity in remote sensing analyses and validate the effectiveness of the resampling and stacking approach implemented in Google Earth Engine.

By this link, my code can be reachable:
https://code.earthengine.google.com/6059dacaed47780ea009305cd204a15b