Prepared by: Juliana Freitas Beyer
Date: 12.12.2025
Thünen Institute of Forestry

Tutorial I: Geodata

Forest Agreement Layer script (version 1.0) tutorial and code explanation

This document provides a concise overview of the Forest Agreement Layer script. It is part of the **GEOS-EUDR** project, which supports the practical implementation of the EU Deforestation Regulation (EUDR) by improving geolocation workflows and the detection of deforestation and forest degradation through remote sensing and geocomputation.

In this context, the term *forest agreement* refers to a multi-source blend of nine global forest and tree-cover datasets. The approach builds on the work of Freitas Beyer et al. (2025), who identified eight datasets as particularly well suited for compliance verification under the EUDR. The current version of the collection extends this foundation by adding a ninth dataset: **Global Forest Types 2020, v0** (Bourgoin et al., 2024). This dataset is available through the JRC portal and the Google Earth Engine catalog:

https://forobs.jrc.ec.europa.eu/GFT and

https://developers.google.com/earth-engine/datasets/catalog/JRC_GFC2020_subtypes_V0

The product remains closely aligned with the criteria defined in the original study by Freitas Beyer et al. (2025). However, because the Global Forest Types 2020 dataset was released after the article's final compilation in November 2024, it could not be included in their assessment.

Table 1. Criteria Compliance Table based on Freitas Beyer et al. (2025)

Short description: The ***Global Forest Types 2020*** **(GFT 2020)** map integrates multiple global datasets that serve as indicators or proxies for the main forest types. It distinguishes three classes - Primary Forest, Naturally Regenerating Forest, and Planted/Plantation Forest, using the 2020 forest extent from version 1 of the *Global Forest Cover* map (GFC 2020) as its base layer. This approach provides a comprehensive, wall-to-wall representation of global forest types for that year.		
Temporal Proximity		2020
Spatial Resolution		10m
Forest Definition	Tree Height	5m
	MMU of Forest Cover	0,5ha
	Forest Canopy Cover	10%
Non-forest tree-based systems included in forest cover?		May contain Agroforestry, however planted trees, "other wooded land" exists only outside the extent of forest cover. the plantation forests are not identified as sub class within planted forests.
Accuracy Metrics		Not explicitly reported. Accuracy only for Global Forest Cover map for year 2020 (GFC 2020)
Map tendency		--

Prepared by: Juliana Freitas Beyer
Date: 12.12.2025
Thünen Institute of Forestry

[T U T O R I A L]

1. INTRODUCTION

This tutorial explains how to generate a forest agreement layer for user-defined geodata polygons or points using Google Earth Engine (GEE). It guides the user through the full workflow from preparing datasets and reclassifying inputs to computing agreement levels, exporting cluster-based outputs, and summarizing forest extent.

The GEE script is organized into **two parts**:

1. a user-defined section where all parameters are set, and
2. an automated section that performs the complete processing sequence.

Script GEODATA – Polygon/Point-based (clustering) version

The **GEODATA script** supports one or multiple polygons/points, such as agricultural plots or management areas. It calculates forest agreement per cluster, where each cluster is formed by a bounding box grouping one or several polygons/points. This structure allows a more targeted analysis across spatially grouped areas.

Usefulness:

- Enables per-cluster visualization of the forest agreement layer and per-polygon/point analysis, making it well suited for site-level assessments.
- Provides clear visual outputs.
- Exports cluster-specific datasets to Google Drive or Earth Engine assets, supporting larger workflows.
- Computes forest extent summaries across all polygons and adds detailed forest-agreement attributes to each production polygon, helping users to compare areas and support spatial decision-making.

2. USER-DEFINED INPUTS

Users can adjust constants controlling spatial resolution, minimum mapping units, forest height thresholds, export behavior and input data settings. These values determine how the processing pipeline behaves and how results are exported.

PART 0: USER-DEFINED CONSTANTS	
var TARGET_RESOLUTION = 30	Spatial resolution in meters of the final forest agreement layer to be exported (the cluster regions). Higher spatial resolution requires more processing power and increases the overall runtime of the script.

var SIEVE_THRESHOLD_PIXELS = 6	Removes small patches below a minimum mapping unit. The advised value is 6 pixels which is equivalent to 0.5ha, a common minimum mapping unit adopted in global maps.
var FOREST_HEIGHT_MIN = 5	This parameter defines the minimum height used to classify forest in height-based tree cover maps. The selected value of 5 meters aligns with the FAO forest definition, which is also adopted in the EUDR regulation.

var AGREEMENT_RADIUS = 1	Defines the size of the neighborhood used to smooth small inconsistencies. For example, setting the value to 1 applies the filter to a 3×3-pixel window, that is, the central pixel plus its eight immediate neighbors in all directions.
	A value of radius = 2 → 5x5 window;
	radius = 3 → 7x7 window, etc.

PART 0A: INPUT SETTINGS (LOAD DATA) A dataset (e.g., shapefile) must be uploaded to GEE Assets before running the script.	
var GEODATA_TYPE = 'Polygon' //'Point'	Select the geometry type of your production data: use “Polygons” directly or let “Points” be expanded into analysis buffers.
var BUFFER_HA = 0.6	Set the minimum area for buffering geodata points. This ensures that only areas above your chosen threshold are included. Note: A practical lower limit for area selection is anything above 0.5 ha. This follows the FAO forest definition, where 0.5 ha is the minimum mapping area of forest.
Var SHAPEFILE_PATH = 'projects/ee-yourusername/assets/your_shapefile_here';	Specify the path for your desired production polygons/point-buffers. The geodata can be in following formats: shapefile, KML/KMZ, .json or csv with geometry. For uploading geodata see “Shapefile Upload” information below.
PART 0B: EXPORT SETTINGS	
var EXPORT_TARGET = 'Drive' //’Asset’	Set to 'Drive' to export to Google Drive or 'Asset' to export to an Earth Engine asset.

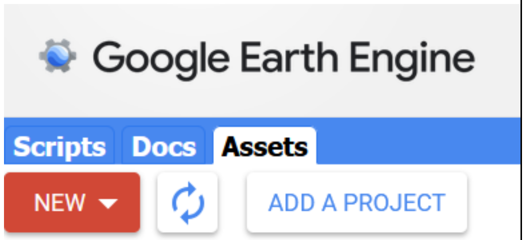
var EXPORT_ASSET_ID = 'users/your_username/ForestAgreement_2020'	Defines the asset path where the exported results will be saved. (In case "Asset" is chosen in "EXPORT_TARGET")
var EXPORT_FOLDER = 'ForestAgreementExports'	Name of the Google Drive folder chosen by user. (In case "Drive" is chosen in "EXPORT_TARGET")
var EXPORT_DESCRIPTION = 'ForestAgreement_2020'	Name assigned to the exported results specified by user.
var EXPORT_FORMAT = 'SHP'	Choose the file format for your exported geodata results. Available options are SHP, KML, KMZ, or GeoJSON.
Note: The Forest Agreement layer exported for each cluster is always provided in **GeoTIFF** format.	

#Shapefile Upload

A Geodata must be uploaded to GEE Assets before running the script.

Steps:

1. Open the Google Earth Engine Code Editor.
2. Go to Assets → NEW → *chose the file format you wish to upload*.
3. Upload the file (e.g., if shapefile upload all .shp, .dbf, .shx, .prj files in compressed form).
4. Copy the resulting asset path and replace the placeholder in SHAPEFILE_PATH.



More information on GEE importing (uploading) shapefiles in Assets:
https://developers.google.com/earth-engine/guides/table_upload

3. AUTOMATIC STEPS

[The following steps are run automatically]

1. Buffering and Cluster-Based Region of Interest

The script generates a user-defined buffer around point features, and then merges nearby polygons or buffered points to form coherent clusters. Bounding boxes are calculated around each cluster to simplify computation and allow exporting each cluster separately. (See figure 1)

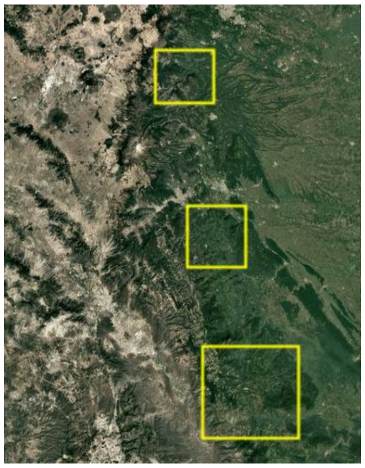


Figure 1. Clusters created over nearby polygons/ point-buffers (groups)

2. Forest Class Definitions

Sets the values corresponding to “Forest” or “Tree cover” class based on dataset guidelines and documentation.

3. Setting of Processing Functions

Binary forest/non-forest reclassification: Converts each dataset into a binary map where forest classes are assigned a value of 1 and all other classes are 0, creating a standardized “Landcover” layer for further analysis.

Reproject and resample: Each dataset is standardized to a consistent resolution (var TARGET_RESOLUTION) and projection. This step ensures pixel-aligned agreement calculations across all datasets.

Filter and mosaic: It filters the image collection by spatial extent and optional date range, then combines the filtered images into a single mosaic and clips it to the region of interest (polygons/point-buffers).

4. Load Datasets

Nine global forest datasets are loaded and clipped to the region of interest (the clusters) to reduce processing load in subsequent steps. **The target year is 2020**, although the actual reference year may vary among datasets due to differences in data availability.

5. Reclassification and Reprojection

Applies the functions defined in Step 3.

6. Forest Agreement Computation

All reclassified forest masks are added together to create a forest-agreement layer that ranges from 0 to 9. Afterwards, a spatial filter is applied to smooth the result, removing small isolated patches and adjusting each pixel based on the prevailing class in its surrounding area.

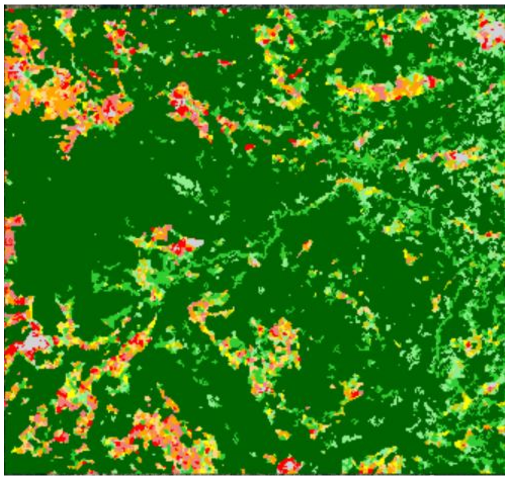


Figure 3. Filtered forest agreement layer

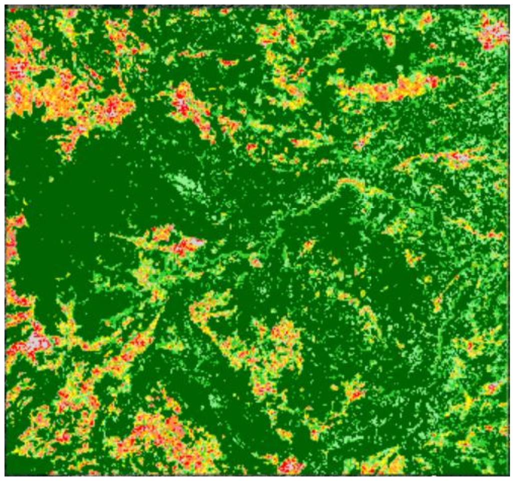


Figure 2. Unfiltered forest agreement layer

7. Visualization by Cluster

To prevent memory overload, only one cluster (the first) is displayed. The map view includes both the raw and the filtered agreement layers, together with the cluster boundaries and the original polygons/point-buffer. A legend helps interpret the results by showing agreement sum from 0 to 9.

<u>Note:</u> only first cluster is displayed, however the export of the forest agreement layers is available for all clusters.

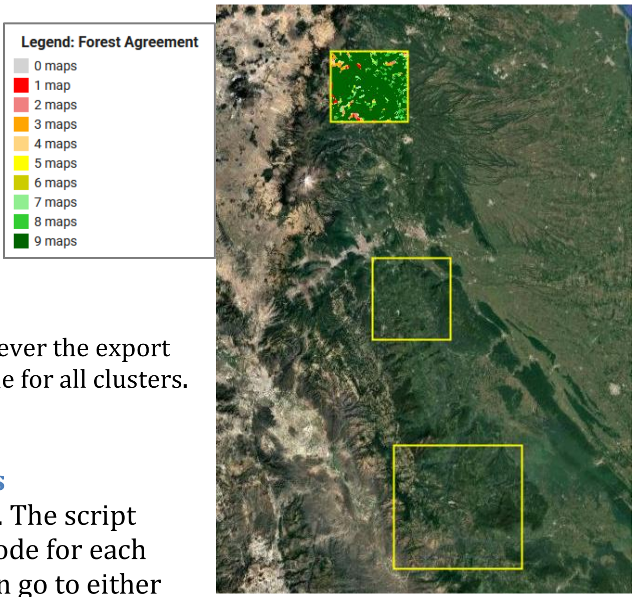


Figure 4. Only 1 cluster (the first) is visualized

8. Exporting Cluster-Based Results

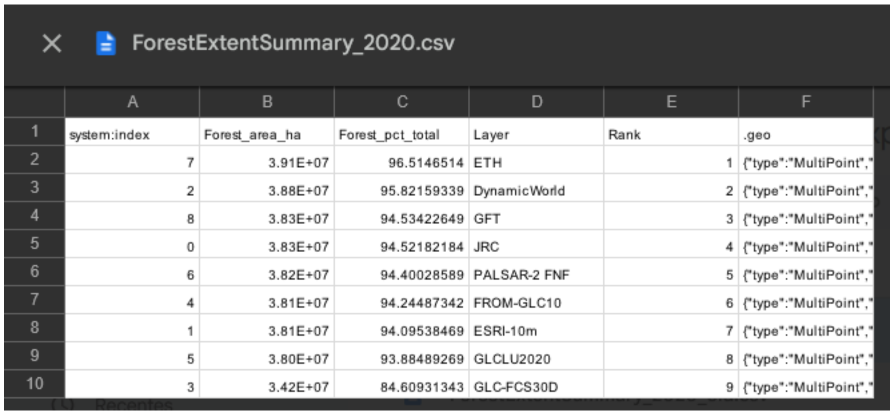
Each cluster is exported individually. The script determines the optimal UTM EPSG code for each cluster using its centroid. Exports can go to either <u>Google Drive</u> or <u>Earth Engine Assets,</u> depending on the option selected in the User Defined Inputs at the beginning of the script.

9. Forest Extent Summary

The script computes the total forest area (in hectares) for each dataset across all polygons/ point-buffers. The datasets are then ranked according to the proportion of forest they map relative to the combined area of all polygons/point-buffers. This provides a way to identify which datasets report the highest or lowest forest cover within the area of interest (i.e., all given geodata polygons/point-buffers).

Forest area is derived from the reclassified binary maps produced for each dataset (see Steps 3 and 5). In the ranking, a value of 1 indicates the dataset with the highest mapped forest cover, while 9 marks the one with the lowest. Results are exported as a CSV file.

CSV files can be opened in Microsoft Excel using Data → From Text/CSV. Please note that the "**Forest_pct_total**"column may be misinterpreted depending on your system's decimal and thousand separators. If the values appear unrealistic, divide the column by 100 to obtain the correct percentages.

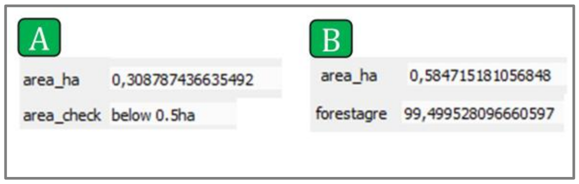

ForestExtentSummary_2020.csv

	A	B	C	D	E	F
1	system:index	Forest_area_ha	Forest_pct_total	Layer	Rank	.geo
2	7	3.91E+07	96.5146514	ETH	1	{"type":"MultiPoint","
3	2	3.88E+07	95.82159339	DynamicWorld	2	{"type":"MultiPoint","
4	8	3.83E+07	94.53422649	GFT	3	{"type":"MultiPoint","
5	0	3.83E+07	94.52182184	JRC	4	{"type":"MultiPoint","
6	6	3.82E+07	94.40028589	PALSAR-2 FNF	5	{"type":"MultiPoint","
7	4	3.81E+07	94.24487342	FROM-GLC10	6	{"type":"MultiPoint","
8	1	3.81E+07	94.09538469	ESRI-10m	7	{"type":"MultiPoint","
9	5	3.80E+07	93.88489269	GLCLU2020	8	{"type":"MultiPoint","
10	3	3.42E+07	84.60931343	GLC-FCS30D	9	{"type":"MultiPoint","

Figure 5. Illustration of the forest extent summary generated by the script (example)

10. Forest Agreement per Polygon/Point-buffer

Forest agreement is calculated as the percentage of the polygon or point-buffer area that falls within the agreement classes defined by 6 to 9 datasets (i.e., the majority of maps). For polygons, a preliminary filter removes units smaller than 0.5 ha to meet the minimum mapping unit used in the FAO forest definition adopted under the EUDR Standard. For points, no filtering is required, as each buffer already represents an area larger than 0.5 ha (see BUFFER_HA). The resulting values are added as attributes to each geometry and exported in the same format as assigned in the EXPORT_FORMAT variable. The added attributes include "forestagree", "area_ha", and "area_check". The "forestagree" field reflects the previously mentioned percentage.



area_ha	0,308787436635492
area_check	below 0.5ha

area_ha	0,584715181056848
forestagre	99,499528096660597

Figure 6. Examples of the resulting forestagree column. (A) The polygon is too small to be considered (< 0.5 ha); (B) The polygon is large enough and shows 99% forest-agreement coverage, meaning that at least six datasets classify almost the entire polygon area as forest.

4. RUNNING THE WORKFLOW

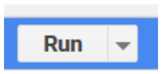
Instructions:

1. Upload a shapefile (or KML, KMZ, or GeoJSON) and update the SHAPEFILE_PATH.

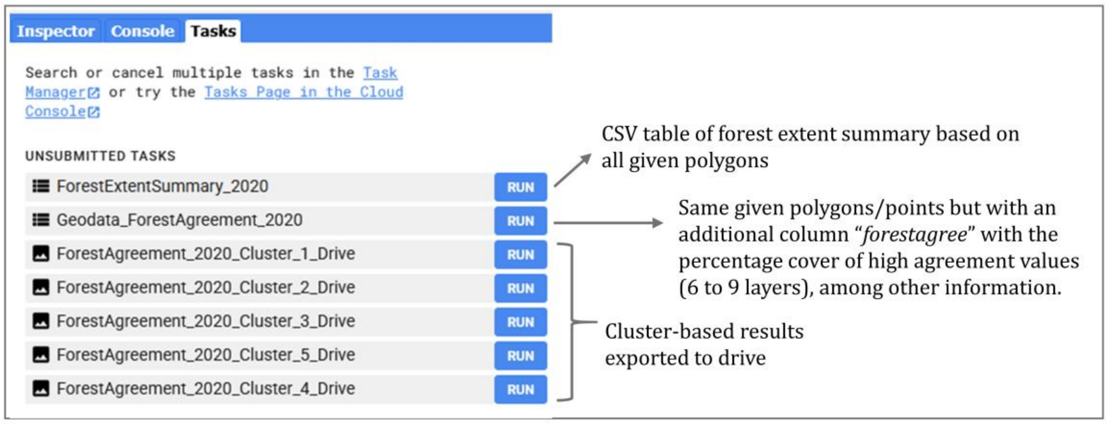
3. Adjust user-defined constants as needed.

4. Set the appropriated input and output settings.

5. **Run** the script by clicking the *Run* button at the top of the Code Editor. -->



6. Monitor exports under the **Tasks** tab.



Search or cancel multiple tasks in the Task Manager or try the Tasks Page in the Cloud Console

UNSUBMITTED TASKS

ForestExtentSummary_2020 — RUN
Geodata_ForestAgreement_2020 — RUN
ForestAgreement_2020_Cluster_1_Drive — RUN
ForestAgreement_2020_Cluster_2_Drive — RUN
ForestAgreement_2020_Cluster_3_Drive — RUN
ForestAgreement_2020_Cluster_5_Drive — RUN
ForestAgreement_2020_Cluster_4_Drive — RUN

CSV table of forest extent summary based on all given polygons

Same given polygons/points but with an additional column "*forestagree*" with the percentage cover of high agreement values (6 to 9 layers), among other information.

Cluster-based results exported to drive

Figure 7. Export results in task

A color legend is provided with this document for users who wish to replicate the GEE colors in GIS software or other geospatial applications. The legend is available in three formats: .txt, .clr, and .xlsx on the GitHub Repository.

For further information and access to data see the GitHub repository: https://github.com/GEOS-EUDR/gee_forest_agreement_layer/tree/main