Prepared by: Juliana Freitas Beyer
Date: 09.01.2025
Thünen Institute of Forestry

Tutorial II: ROI (defined region of interest)

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)

<table>
<tr><td colspan="3">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.</td></tr>
<tr><td colspan="2">Temporal Proximity</td><td>2020</td></tr>
<tr><td colspan="2">Spatial Resolution</td><td>10m</td></tr>
<tr><td rowspan="3">Forest Definition</td><td>Tree Height</td><td>5m</td></tr>
<tr><td>MMU of Forest Cover</td><td>0,5ha</td></tr>
<tr><td>Forest Canopy Cover</td><td>10%</td></tr>
<tr><td colspan="2">Non-forest tree-based systems included in forest cover?</td><td>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.</td></tr>
<tr><td colspan="2">Accuracy Metrics</td><td>Not explicitly reported. Accuracy only for Global Forest Cover map for year 2020 (GFC 2020)</td></tr>
<tr><td colspan="2">Map tendency</td><td>--</td></tr>
</table>

[T U T O R I A L]

1. INTRODUCTION

This tutorial explains how to generate a forest agreement layer for **user-defined region of interest (ROI)** using Google Earth Engine (GEE). It guides the user through the full workflow from preparing datasets and reclassifying inputs to computing agreement levels, exporting tiled-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 ROI – Defined region of interest version

The **ROI script** evaluates forest agreement within a single user-defined region of interest, such as an administrative boundary. It computes the agreement for the entire area, but the outputs are exported in smaller tiles to keep the processing in GEE efficient.

Usefulness:

- Enables rapid assessment of forest agreement for a specific region.
- Generates a visual map (buffer) with color-coded agreement levels, aiding interpretation.
- Exports the full dataset in manageable tiles, making large-area analysis feasible in Earth Engine.
- Produces a summary table of forest extent across datasets, supporting quantitative comparison and reporting.

The ROIs can be defined through:

- Drawing directly in the Code Editor
- Using built-in administrative or thematic boundaries
- Importing a geometry from a Fusion Table (legacy) or Google Drive/Assets
- Creating a geometry programmatically
- Using the bounds of an image or collection, etc.

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.
var VIS_BUFFER = 30000	Distance, in meters, added as a buffer around the centroid of the ROI to define the visualization window.
PART 0A: INPUT SETTINGS (LOAD DATA) *(commented out by default)* -> Uncomment this section if you prefer to define your ROI using **another method instead of drawing geometry** directly in the Code Editor. -> Choose the code line that matches your preferred input method and remove the "/*" and "*/" before and after the code lines to activate it. -> For more information on how to draw a geometry in code editor see section "**Defined ROI**" in "Automatic Steps"	
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")

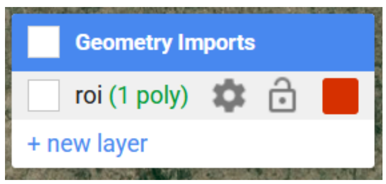
var EXPORT_FOLDER = 'ForestAgreementExports'	Name of the Google Drive folder chosen by user. (In case "Drive" is chosen in "EXPORT_TARGET")
var EXPORT_DESCRIPTION = 'ForestAgreement_2020'	Name assigned to the exported results specified by user.
Note: The Forest Agreement layer exported for each cluster is always provided in **GeoTIFF** format.	

3. AUTOMATIC STEPS

[*The following steps are run automatically***]**

1. Defined ROI

This section displays the defined ROI. In case it is not defined, an error message will appear, asking user to defined it.



The script can only be run with a defined "roi". See figure 1 and 2.

Figure 1. Example of a ROI defined through drawing directly in the code editor

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.

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.

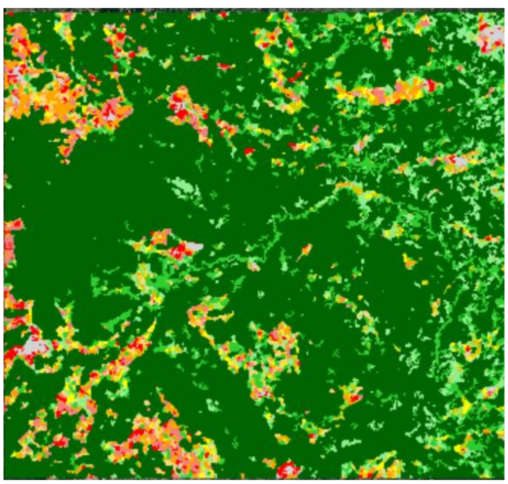


Figure 3. Filtered forest agreement layer

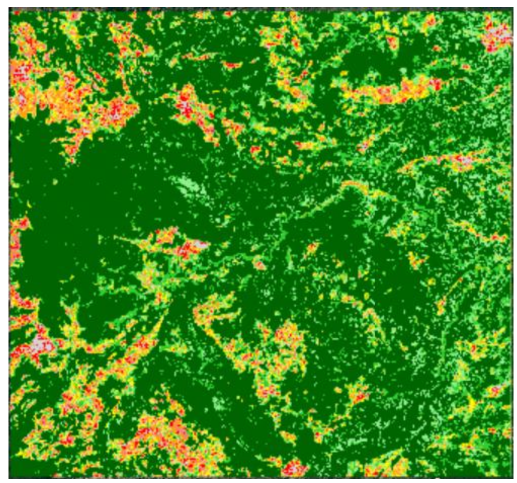


Figure 2. Unfiltered forest agreement layer

7. Visualization of buffer at centroid

This section creates a simplified map view by focusing on a buffered area around the ROI's centroid and using a downscaled resolution for faster rendering. Both the filtered and raw agreement layers are reprojected, clipped, and displayed with a shared color palette that represents agreement values from 0 to 9. A custom legend is added to guide interpretation, and the map is centered on the buffered visualization window for a clear and lightweight overview.

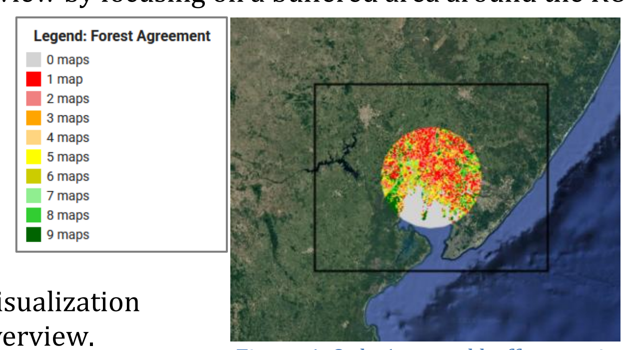


Figure 4. Only 1 central buffer area is visualized

Note: Only the buffer is displayed, however the export of the forest agreement layers is available for the whole ROI.

8. Exporting Tile-Based Results

This part prepares the region of interest for export by dividing it into a grid of tiles and exporting each tile individually. First, the code identifies the most suitable UTM projection based on the ROI's location. It then calculates the geographic extent, splits it into rows and columns according to the chosen tiling scheme, and creates a feature collection representing each tile. The filtered agreement layer is clipped to the ROI once, and each tile is exported with its own tailored projection, either to Google Drive or an Earth Engine Asset. The tiling approach makes the export process more reliable and manageable for large areas.

9. Forest Extent Summary

The script computes the total forest area (in hectares) for each dataset across the entire ROI. The datasets are then ranked according to the proportion of forest they map relative to the full ROI extent. This provides a straightforward way to identify which datasets report the highest or lowest forest cover within the ROI.

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 mapped forest cover. 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

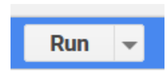
	A	B	C	D	E	F
1	system:index	Forest_area_ha	Forest_pct_total	Layer	Rank	.geo
2	7	3.91E+07	96.5146514	ETH	1	{"type":"MultiPoint","
3	2	3.88E+07	95.82159339	DynamicWorld	2	{"type":"MultiPoint","
4	8	3.83E+07	94.53422649	GFT	3	{"type":"MultiPoint","
5	0	3.83E+07	94.52182184	JRC	4	{"type":"MultiPoint","
6	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)

4. RUNNING THE WORKFLOW

Instructions:

1. Define a ROI through different methods.
3. Adjust user-defined constants as needed.
4. Set the export settings.

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

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

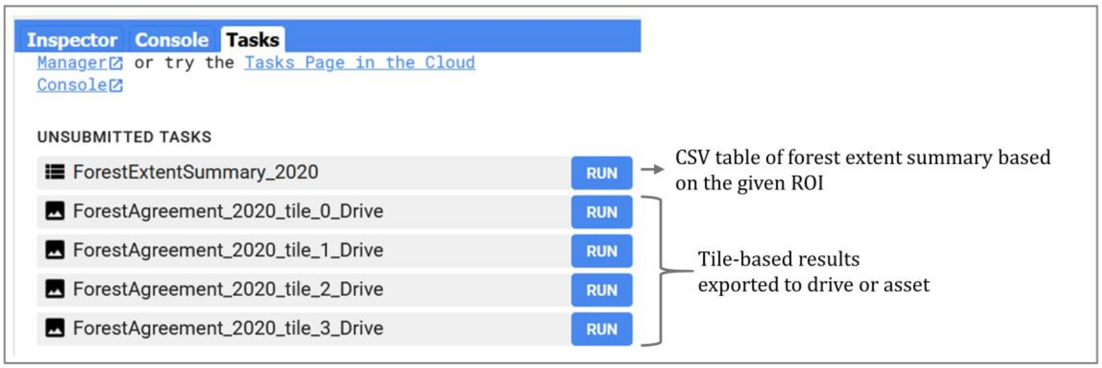


Figure 6. Export results in task

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