VHI calculation Google Earth Engine open-source code

(developed by authors for calculating the mean day and night LST during summer for the study area)

Northeast Italian viticulture affected by heat and vegetation stress. A Satellite-Based Study from 2000 to 2024

Vincenzo Baldan, Eugenio Straffelini, Vincenzo D’Agostino, and Paolo Tarolli (2025)

//import study area boundaries

//import MODIS LST and NDVI dataset

var LST = ee.ImageCollection("MODIS/061/MOD11A2").select('LST\_Day\_1km');

var NDVI = ee.ImageCollection("MODIS/061/MOD13Q1").select('NDVI');

//define the period you want to base the following proceedings

var historical\_start = ee.Date("2000-01-01");

var historical\_end = ee.Date("2024-12-31");

// Define weighting factor

var a = 0.5; // Weight for VCI

// Compute historical min/max for NDVI and LST

var NDVI\_hist = NDVI.filterDate(historical\_start, historical\_end);

var LST\_hist = LST.filterDate(historical\_start, historical\_end);

var IminNDVI = NDVI\_hist.reduce(ee.Reducer.min());

var ImaxNDVI = NDVI\_hist.reduce(ee.Reducer.max());

var IminLST = LST\_hist.reduce(ee.Reducer.min());

var ImaxLST = LST\_hist.reduce(ee.Reducer.max());

// Filter data for summer for each year of the period considered

var NDVI\_summer = NDVI.filter(ee.Filter.calendarRange(2000, 2004, 'year'))

.filter(ee.Filter.calendarRange(6, 8, 'month'));

var LST\_summer = LST.filter(ee.Filter.calendarRange(2000, 2004, 'year'))

.filter(ee.Filter.calendarRange(6, 8, 'month'));

// Compute VCI

var VCI = NDVI\_summer.map(function(image) {

return image.expression('(Ia - Imin) / (Imax - Imin) \* 100', {

Ia: image,

Imin: IminNDVI,

Imax: ImaxNDVI

}).rename('VCI')

.copyProperties(image, ['system:time\_start']);

});

// Compute TCI

var TCI = LST\_summer.map(function(image) {

return image.expression('(Imax - Ia) / (Imax - Imin) \* 100', {

Ia: image,

Imin: IminLST,

Imax: ImaxLST

}).rename('TCI')

.copyProperties(image, ['system:time\_start']);

});

// Join VCI and TCI by time

var filter = ee.Filter.equals({leftField: 'system:time\_start', rightField: 'system:time\_start'});

var join = ee.Join.inner();

var joined = join.apply(VCI, TCI, filter);

//calculate VHI

var VHI = ee.ImageCollection(joined.map(function(feature) {

var vci = ee.Image(feature.get('primary'));

var tci = ee.Image(feature.get('secondary'));

return vci.addBands(tci)

.expression('a \* VCI + (1 - a) \* TCI', {

'VCI': vci.select('VCI'),

'TCI': tci.select('TCI'),

'a': a

}).rename('VHI')

.copyProperties(vci, ['system:time\_start']);

}));

print(VHI)

//get the mean VHI

var meanVHI = VHI.mean();

Map.addLayer(meanVHI, {min: 0, max: 30, palette: ['red', 'orange', 'yellow', 'green']}, 'meanVHI');

//Calculate the frequency distribution of VHI < 40

var frequency\_VHI\_40 = VHI.map(function(image) {

return image.lt(40).rename('VHI\_below\_40')

.copyProperties(image, ['system:time\_start']);

}).sum().clip(table);

var frequency\_VHI\_40\_float = frequency\_VHI\_40.float()

// Display results

Map.addLayer(frequency\_VHI\_40, {min: 0, max: 30, palette: ['#ffffcc', '#ffcc00', '#ff3300', '#990000']}, 'Frequency VHI < 40');

// Export maps

Export.image.toDrive({

image: meanVHI, //or frequency\_VHI\_40

description: 'VHI\_mean\_0004',

region: table, //study area boundaries

scale: 250,

crs: 32632,

maxPixels: 1e13

});