Supplementary Information:

**Deriving wetland-cover types (WCTs) from the integration of multispectral indices based on Earth Observation data**

**Manudeo Singh1,2, Satyasri Allaka1, Praveen K. Gupta3, J.G. Patel3 and Rajiv Sinha1**

1Department of Earth Sciences, Indian Institute of Technology Kanpur

2Department of Earth and Environmental Sciences, University of Potsdam, Germany

3Space Applications centre, ISRO Ahmedabad

## **S1: Method**

A total of 23 indices which are sensitive to the optical properties of waterbodies, especially wetlands, were calculated using multispectral dataset (Table S1). To find the best possible combination of indices for wetland cover type (WCT) classification, we used statistical methods Principal Component Analysis (PCA) and Detrended Correspondence Analysis (DCA) (Fig. S1 and S2). We found that three indices NDVI, NDWI, and NDTI are well distanced from each other in both PCA and DCA plots. Further, these three indices correspond to three different biophysical constituents’ vegetation, water, and turbidity in an aquatic environment. To further ascertain the applicability of these three indices for WCT classification, we performed an ISO-clustering with all 23 indices and another ISO-clustering with only these 3 indices NDVI, NDWI, and NDTI (Fig. S3). We observed that there 3 indices are able to capture the surface variabilities as good as the 23 indices. Based on these assessments, we concluded that the three indices NDWI, NDVI, and NDTI are well suited for the WCT classification of wetlands.

Table S1: Spectral indices used for optically sensitive water constituents.

|  |  |  |  |
| --- | --- | --- | --- |
| Optical parameters | Spectral indices | Formulas | References |
| Water | NDWI |  | (Ji et al., 2009; McFeeters, 1996) |
| NDPI |  | (Lacaux et al., 2007) |
| MNDWI |  | (Xu, 2006) |
| Chlorophyll | NDVI |  | (Townshend and Justice, 1986; Tucker and Sellers, 1986) |
| NDMI (Moisture) |  | (Rokni et al., 2014) |
| NDCI |  | (Mishra and Mishra, 2012) |
| MCI |  | (Gower et al., 2008) |
| Red tide index |  | (Ahn and Shanmugam, 2006) |
| Band difference |  | (Gholizadeh et al., 2016; Ha et al., 2017; Matthews, 2011) |
| Band ratios |  |
|  |
|  |
| Salinity | NDSI |  | (Odeh and Onus, 2008) |
| Turbidity | NDTI |  | (Ji et al., 2009; Lacaux et al., 2007) |
| Secchi Disk Depth | Band ratio |  | (Gholizadeh et al., 2016; Matthews, 2011) |
| CDOM | Band ratio |  |
| Suspended Matter | NDMI (Material) |  | (Arisanty and Saputra, 2017) |
| NDSSI |  | (Hossain et al., 2010) |
| Band ratio |  | (Gholizadeh et al., 2016; Matthews, 2011) |
|  |

**NDPI –** Normalized Difference Pond Index, **MNDWI –** Modified Normalized Difference WaterIndex, **NDMI –** Normalized Difference Moisture Index**, NDCI –** Normalized Difference Chlorophyll Index, **MCI –** Maximum Chlorophyll Index, **NDSI –** Normalized Difference Salinity Index, **NDMI –** Normalized Difference Material Index, **NDSSI –** Normalized Difference Suspended Sediment Index. B1, B2, B3, B4, B5, B8, and SWIRS represent the reflectance of coastal aerosol, Blue, Green, Red, NIR and SWIR (SWIR1:B11 and SWIR 2:B12) bands of the Sentinel-2 datasets respectively.

Chart

Description automatically generated

Fig. S1: PCA of 23 multispectral indices.

A picture containing text, indoor

Description automatically generated

Fig. S2: DCA of 23 multispectral indices.

Map

Description automatically generated

Fig. S3: ISO-clustering using (a) 23 multispectral indices, and (b) three WCT indices – NDWI, NDVI, NDTI into 10 classes. NOTE: Colour codes are not meaningful and similar colours in a and b do not represent similar underlying bio-physical property.

## **S2: Google Earth Engine (GEE) application for automatic WCT calculation**

The WCT classification algorithm is implemented in the GEE. A link to the code and the actual code is presented here. The outputs from the GEE do not contain an alphanumeric WCT coding scheme as presented in the manuscript (Table 1) but as integer keys. A correspondence between GEE integer keys and alphanumeric keys of WCTs is presented below (Table S2). We have provided an ArcGIS tool to automatically convert GEE integer keys into alphanumeric keys (presented in next section).

Table S2: Alphanumeric GEE keys and their corresponding GEE integer keys in parentheses.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thresholds** | | **Unique IDs** | | |
| **Values** | **Range** | **NDWI** | **NDVI** | **NDTI** |
| 0.0 - 0.25 | Low | W1 (110000) | V1 (1100) | T1 (11) |
| 0.25 – 0.50 | Medium | W2 (220000) | V2 (2200) | T2 (22) |
| 0.50 – 0.75 | High | W3 (330000) | V3 (3300) | T3 (33) |
| 0.75 – 1.0 | Very High | W4 (440000) | V4 (4400) | T4 (44) |

**Link of code:** <https://code.earthengine.google.com/9664d4eeaa1e84fd251c0370b871b498>

**Full code:**

//Wetland polygon

var table = ee.FeatureCollection("users/manudeo\_singh/Chilika\_Lagoon");

//Date bounds

var startDate = '2020-10-01';

var endDate = '2020-11-15';

//Get satallite imageries collection for required dates and region (wetland) and apply a simple cloud filter

var data= ee.ImageCollection('COPERNICUS/S2\_SR')

.filterDate(startDate,endDate)

.filterBounds(table)

.filter(ee.Filter.lt('CLOUDY\_PIXEL\_PERCENTAGE',20));

//Calculate median values of bands for given date range in the collection

var data\_med = data.reduce(ee.Reducer.median());

//print(data\_med);

Map.centerObject(table, 10);

//Map.addLayer(table); //visualise wetland polygon

//Clip satellite imagery raster to wetland region and visualise the FCC

var input\_raster = data\_med.clip(table);

Map.addLayer(input\_raster, {bands: ['B8\_median', 'B3\_median', 'B4\_median'], min: 500, max: 3500});

//Calculate WCT indices

var NDWI= input\_raster.normalizedDifference(['B3\_median','B8\_median']);

var NDVI= input\_raster.normalizedDifference(['B8\_median','B4\_median']);

var NDTI= input\_raster.normalizedDifference(['B4\_median','B3\_median']);

//Reclass WCT indices in four classes

//Reclass NDWI

var NDWI\_Reclass = NDWI

.where(NDWI.lte(0), 0)

.where(NDWI.gt(0).and(NDWI.lte(0.25)), 110000)

.where(NDWI.gt(0.25).and(NDWI.lte(0.5)), 220000)

.where(NDWI.gt(0.5).and(NDWI.lte(0.75)), 330000)

.where(NDWI.gt(0.75), 440000);

//Reclass NDVI

var NDVI\_Reclass = NDVI

.where(NDVI.lte(0), 0)

.where(NDVI.gt(0).and(NDVI.lte(0.25)), 1100)

.where(NDVI.gt(0.25).and(NDVI.lte(0.5)), 2200)

.where(NDVI.gt(0.5).and(NDVI.lte(0.75)), 3300)

.where(NDVI.gt(0.75), 4400);

//Reclass NDTI

var NDTI\_Reclass = NDTI

.where(NDTI.lte(0), 0)

.where(NDTI.gt(0).and(NDTI.lte(0.25)), 11)

.where(NDTI.gt(0.25).and(NDTI.lte(0.5)), 22)

.where(NDTI.gt(0.5).and(NDTI.lte(0.75)), 33)

.where(NDTI.gt(0.75), 44);

//Calculate WCTs

var wct\_int = (

NDWI\_Reclass

.add(NDVI\_Reclass)

.add(NDTI\_Reclass)

.rename('WCT'))

.int(

);

// Get the date range of images in the collection.

var range = data.reduceColumns(ee.Reducer.minMax(),

["system:time\_start"]);

var time = ('Date range: ', [ee.Date(range.get('min')),

ee.Date(range.get('max'))]);

print(time);

//Get the list of WCTs

var reduction = wct\_int.reduceRegion({

reducer:ee.Reducer.frequencyHistogram(),

geometry: table,

scale: 10

});

var wctVals = ee.Dictionary(reduction.get(wct\_int.bandNames().get(0)))

.keys()

.map(ee.Number.parse);

print(wctVals);

//Visualise the WCT layer

var wctViz = {min: 11, max: 444444, palette: ['orange', 'green', 'blue']};

Map.addLayer(wct\_int, wctViz, 'WCT', true);

//Export output to Google Drive

Export.image.toDrive({

image: wct\_int,

description: 'wct',

scale: 10,

region: table,

maxPixels: 1e9,

fileFormat: 'GeoTIFF'

});

## **S3: ArcGIS tools for automatic WCT calculations**

We developed a set of tools for ArcGIS to automate the WCT generation (Fig. S4). Apart from GEE based WCT calculation, a standalone ArcGIS tool is also developed to calculate the WCT for any sensor with VNIR bands (Figs. S5 and S6). The WCT tools shown in Figs. S5 and S6 require green, red, and NIR bands, a working directory to save the resulting WCT, and the name for the WCT file (without file format, GeoTIFF is the default output format). The first tool among these two is for WCT calculation of a wetland with shoreline polygon. The shoreline polygon is used as mask to calculate WCT only for the wetland region.

Apart from these two standalone WCT calculation tools, two tools are created to convert the GEE calculated WCT with integer WCT keys to alphanumeric WCT keys (Table S2). First tool is for batch processing of multiple rasters. This tool requires the folder path with WCT files. Second tool is for single GEE calculated WCT raster.

For all four tool, RGB colours are automatically added as colourmap fields so that the colour for a given WCT can remains same.

Graphical user interface, text, application, email

Description automatically generated

Fig. S4: ArcGIS tools list for WCT calculations

Graphical user interface, application

Description automatically generated

Fig. S5: ArcGIS tool for WCT calculation for wetlands with shoreline polygon.

Graphical user interface, text, application, email

Description automatically generated

Fig. S6: ArcGIS tool for WCT calculation for wetlands without shoreline polygon.