Harmful Algal Bloom Detection Using MODIS Imagery: Implementation in Google Earth Engine

Maya Midzik

FES 754

Red Tide Detection Using MODIS Imagery: Implementation in Google Earth Engine

Background

Phytoplankton blooms are the base of marine food webs, however a small percentage of algal species produce toxins that kill fish, mammals and birds, as well as causing human illness. Accurate monitoring and forecasting of these harmful algal bloom (HAB) events, also known as red tides, is critical for ecosystem monitoring and the protection of human health.

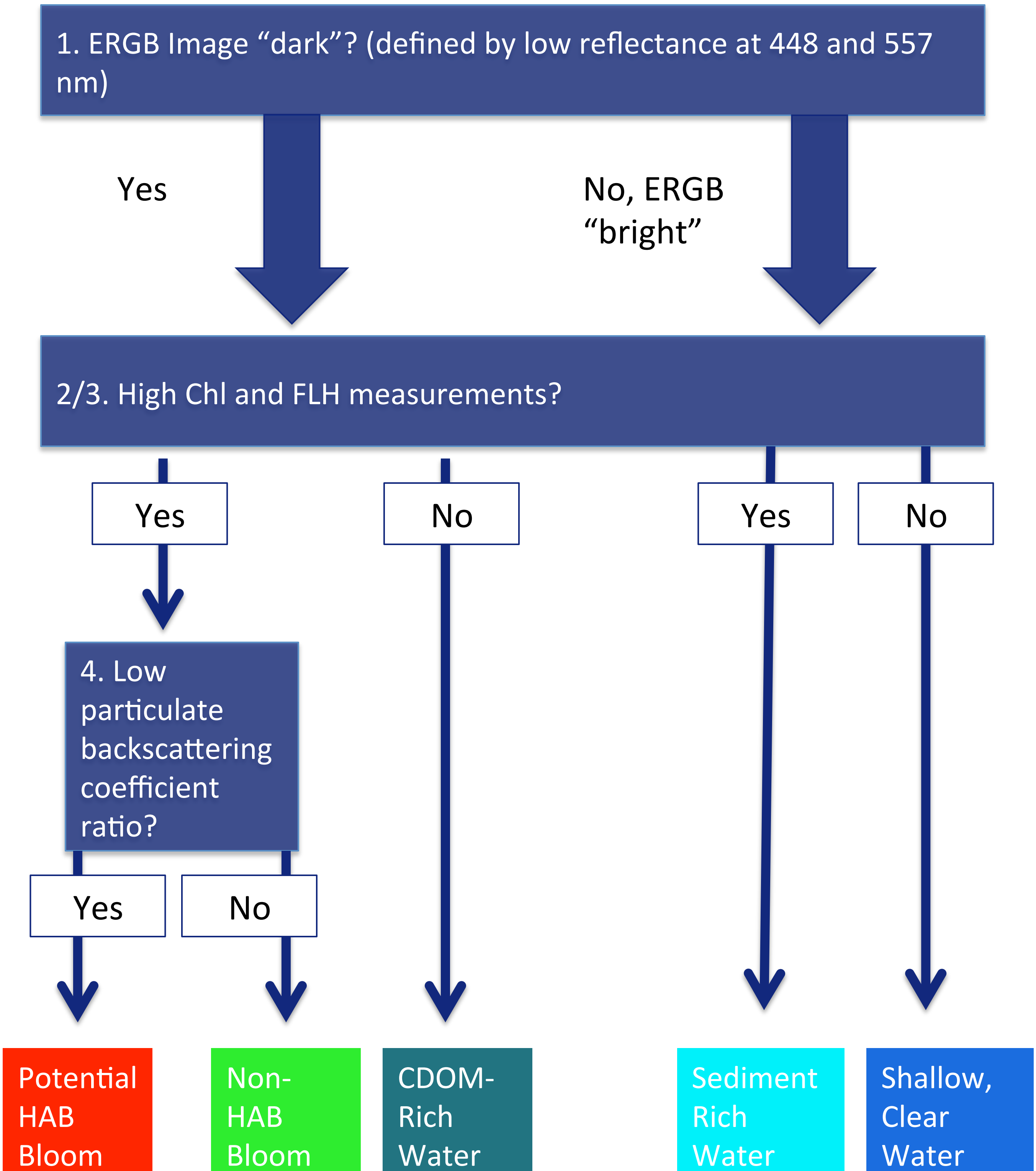
Synoptic and frequent observations are required for forecasting and mitigating HAB blooms, and number of remote sensing models have been proposed to characterize these conditions. This study uses Moderate Resolution Imaging Spectroradiometer (MODIS) satellite imagery and the methods proposed by Hu et al. (2013) to classify water conditions.

HAB Remote Sensing Model

Model Inputs:

1. Enhanced RGB (ERGB) image
   - Distinguishes high Chl-a or CDOM water from suspended sediments and shallow bottoms
2. Chlorophyll a (Chl-a) from an empirical band-ratio algorithm (OC4v4; O'Reilly et al., 2000)
3. Fluorescence Line Height
   - A more accurate Chl measurement. A relative measure of the amount of radiance leaving the sea surface in the chlorophyll fluorescence emission band
4. Particulate backscattering coefficient ratio at 551 nm
   - The ratio of two backscattering coefficients: *b*bp,551/*b*bp,Morel. Red tide blooms exhibit a lower backscatter compared to diatom bloom (Cannizzaro et al. 2002)

Using Model Inputs to Determine Water Conditions



Methods

Pre-processing and data selection

The following initial processing was completed in NASA's SeaWiFS Data Analysis System (SeaDAS) software:

1. MODIS Level 1-a data obtained from NASA Ocean Color Website
2. MODIS/Aqua Level-1a data were processed to generate Level-1b files and SeaDAS using Seadas MODIS_L1b and MODIS_GEO codes
3. MODIS Level-1b files were atmospherically corrected using l2gen code, generating remote sensing reflectance, chl-a, particulate backscattering using the Quasi-Analytical Algorithm (QAA, Lee et al., 2002) and normalized water-leaving radiance for MODIS bands

From Seadas, 4 GeoTiFF files were uploaded to Google Earth Engine containingthe precursors for necessary ocean products (to be calculated and analyzed in GEE):

1. nLw(λ) data at 551, 488, and 443 nm to compose an Enhanced RGB (ERGB) image
2. Remote sensing reflectance (Rrs) at 488 and 547 for OC3 chlorophyll calculations
3. nLw(λ) data from three MODIS wavebands at 667, 678, and 748 nm for Fluorescence Line Height calculation
4. particulate backscattering coefficient at 551 nm (*b*bp,551) using a Quasi-Analytical Algorithm (QAA, Lee et al., 2002)

Generation of Model Inputs in GEE

1. Enhanced RGB (ERGB) image generation:

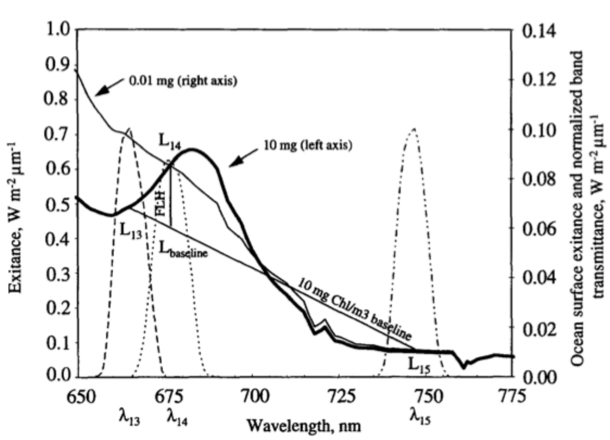
nLw(λ) data at 551, 488, and 443 nm were used as the red, green and blue channels to compose an Enhanced RGB (ERGB) image.

2. Chlorophyll-a determined using the NASA OC3 algorithm

Rrs1 = blue wavelength Rrs (443)
Rrs2 = green wavelength Rrs (547)
X = log10(Rrs1 / Rrs2)
chlor_a = 10^(a0 + a1*X + a2*X^2 + a3*X^3 + a4*X^4)

3. Fluorescence Line height:

Derived using linear baseline algorithm (Letelier et al., 1996)



$$FLH = L_{14} - L_{baseline}$$

$$L_{baseline} = L_{15} + (L_{13} - L_{15}) * \left[\frac{\lambda_{15} - \lambda_{14}}{\lambda_{15} - \lambda_{13}}\right]$$

L13= 667 nm (665.1 center)
L14 = 678 nm (676.7 center)
L15= 748 nm (746.3 center)

4. Particulate Backscattering Coefficient Ratio

A. *b*bp*,*551= estimated with the QAA algorithm (Lee et al., 2002)

B. backscattering coefficient at 551 nm using the Morel (2001) algorithm

*b*bp*,*Morel = 0.3 × Chl0.62 × *(*0.002 + 0.02 × *(*0.5 − 0.25 × log10Chl*))*

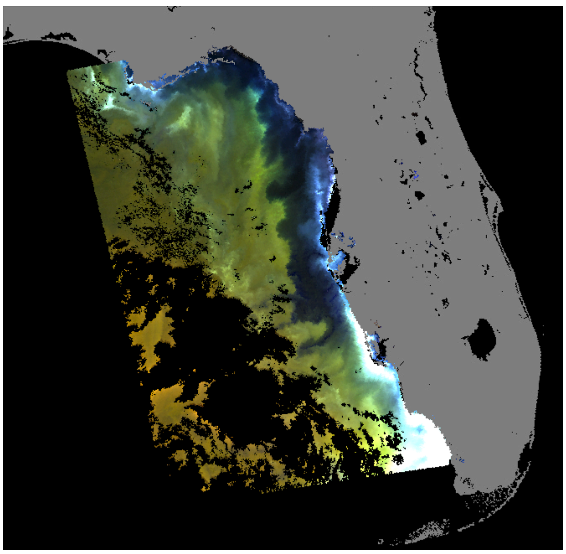
C. Backscattering Coefficient Ratio: *b*bp*,*551/*b*bp*,*Morel

Case Study: Florida Gulf Coast

MODIS Imagery: 21 January, 2005

Registered extensive red tide event along coast at this date (in situ sampling confirms locations)

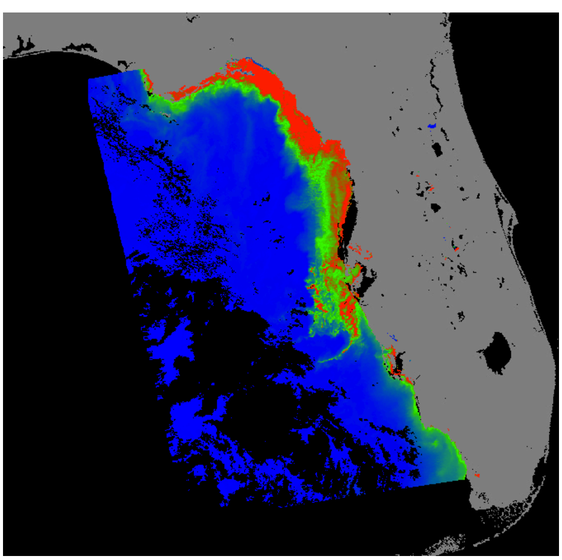
No-data region in bottom left due to cloud-covered area



1. Enhanced RGB (ERGB) image

Dark colors from increased light absorption at the blue wavelength due to high Chl-a or CDOM

Light colors from suspended sediments and shallow bottoms



2. Chlorophyll-a determined using the NASA OC3 algorithm

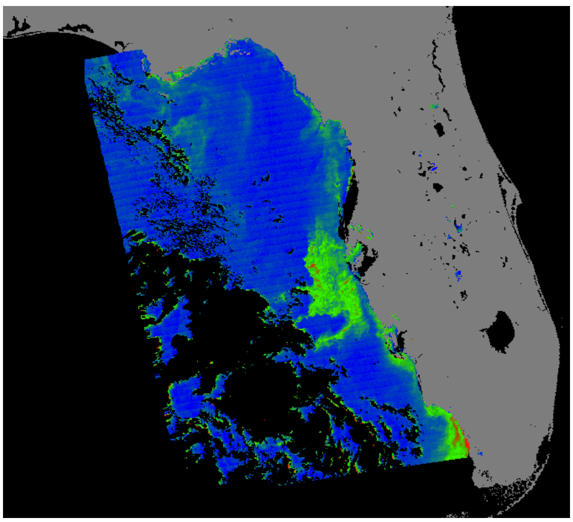
High Chl= red

NASA OC3 algorithm indicates high Chl-a levels along entire coast due to contamination in water-leaving signal

6

Case Study: Florida Gulf Coast

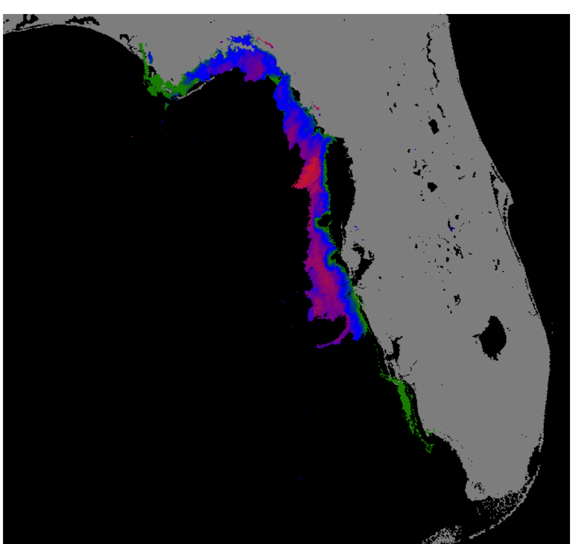
3. Fluorescence Line height:



High FLH= bright green

FLH distinguishes dark CDOM-rich waters (incorrectly interpreted as high Chl-*a* in band ratio algorithms) from phytoplankton-rich water. However, FLH is not reliable in sediment-rich waters, and 'false' postives may occur under these conditions.

4. Particulate Backscattering Coefficient Ratio



Low bbp ratio = bright pink colors
Indicates HAB/ red tide bloom (as opposed to normal phytoplankton bloom

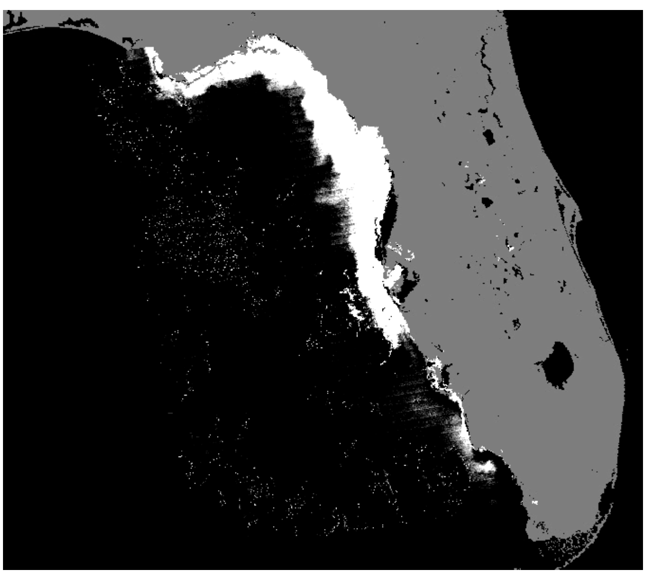
7

Chlorophyll Error

The chlorophyll levels derived from the OC3 algorithm resulted in incorrect/elevated water classification along the entire coast. This is due to the incorrect interpretation of CDOM-rich waters in the algorithm as having high CHl content.

Chlorophyll error was calculated by comparing OC3 derived Chl to FLH measurements

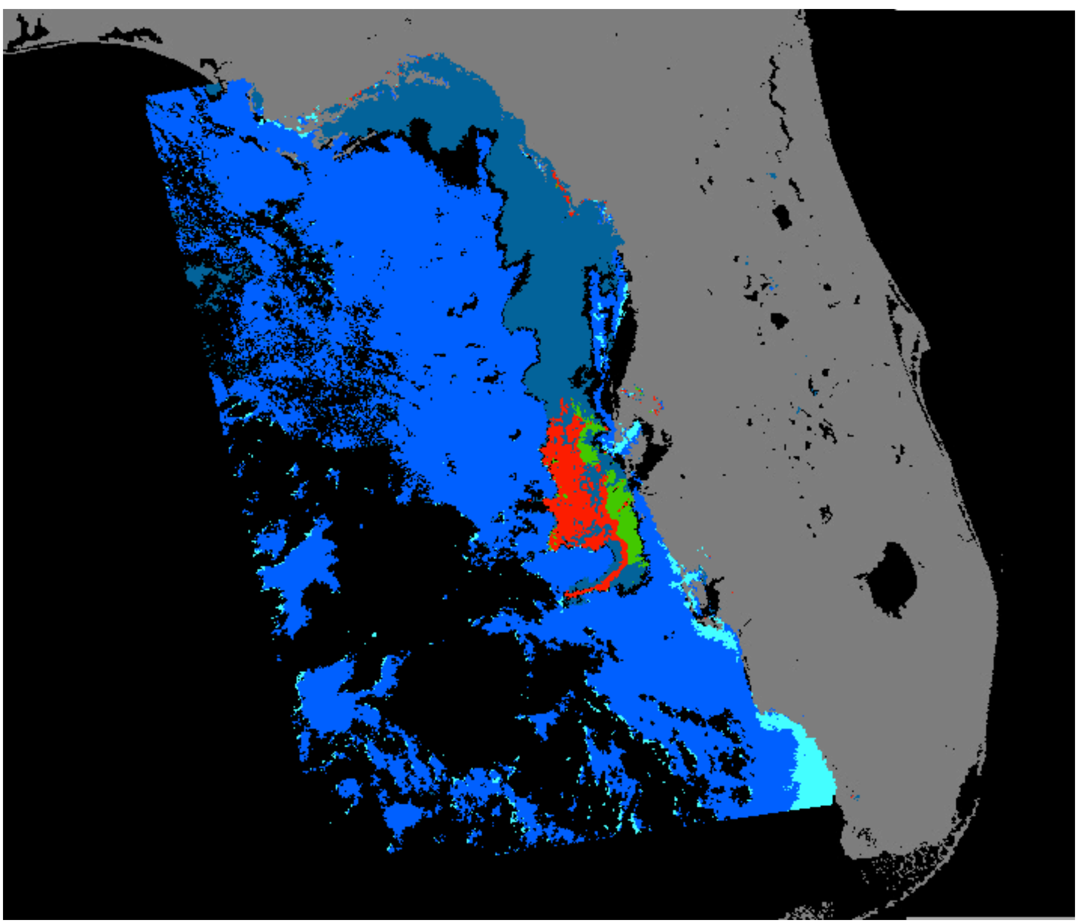
Bright pixels = high Chl error



21 January, 2005

Final Classification Based on Model Inputs

- Potential HAB Bloom
- Non-HAB Bloom
- CDOM-Rich Water
- Sediment Rich Water
- Shallow, Clear Water



21 January, 2005

Conclusions

HAB detection was successfully implemented in Google Earth Engine using SeaDAS-processed MODIS data and techniques proposed by Hu et al (2005, 2013) and Cannizzaro et al. (2008). Although currently HAB detection requires scene-based pre-processing, this study demonstrates the potential of GEE's global computing power for synoptic measurements of harmful algal blooms.

The addition of basic ocean color to the Earth Engine available datasets could prove a powerful tool for large-scale and open ocean and coastal remote sensing.

A note on projection errors: MODIS images processed in SeaDAS and uploaded to GEE were subject to projection errors, leading to incorrect alignment with the Earth Engine basemap. This is a problem is not unique to this dataset, and has been recognized in GIS application of MODIS images due to different Geographic Coordinate and Projected Coordinate Systems (GCS, PCS). A number of scene-based reprojections were attempted in this study, both within SeaDAS and GEE itself with no successful change to map alignment.

Resources

Cannizzaro, Jennifer P., Kendall L. Carder, F. Robert Chen, John J. Walsh, Zhongping Lee, and Cynthia Heil. "A novel optical classification technique for detection of red tides in the Gulf of Mexico." Harmful Algae (2002): 282-434.

Hu, Chuanmin, Frank E. Muller-Karger, Charles Judd Taylor, Kendall L. Carder, Christopher Kelble, Elizabeth Johns, and Cynthia A. Heil. "Red tide detection and tracing using MODIS fluorescence data: A regional example in SW Florida coastal waters." Remote Sensing of Environment 97, no. 3 (2005): 311-321.

Hu, Chuanmin. "A novel ocean color index to detect floating algae in the global oceans." Remote Sensing of Environment 113, no. 10 (2009): 2118-2129.

Harvard

Lee, ZhongPing, Kendall L. Carder, and Robert A. Arnone. "Deriving inherent optical properties from water color: a multiband quasi-analytical algorithm for optically deep waters." Applied optics 41, no. 27 (2002): 5755-5772.

Letelier, Ricardo M., and Mark R. Abbott. "An analysis of chlorophyll fluorescence algorithms for the Moderate Resolution Imaging Spectrometer (MODIS)." Remote Sensing of Environment 58, no. 2 (1996): 215-223.

Morel, André, and Stéphane Maritorena. "Bio-optical properties of oceanic waters: A reappraisal." Journal of Geophysical Research: Oceans (1978–2012) 106, no. C4 (2001): 7163-7180.

O'Reilly, John E., Stephane Maritorena, David A. Siegel, Margaret C. O'Brien, Dierdre Toole, B. Greg Mitchell, Mati Kahru et al. "Ocean color chlorophyll a algorithms for SeaWiFS, OC2, and OC4: Version 4." SeaWiFS postlaunch calibration and validation analyses, Part 3 (2000): 9-23.

GEE Code

```
//Harmful Algal Bloom Analysis from MODIS
//Maya Midzik: mmidzik@gmail.com
//FES 754

/* This  code differentiates ride tides/harmful algal blooms (HABs)
from other near coastal waters and phytoplankton blooms
using   Moderate Resolution Imaging Spectroradiometer (MODIS)
satellite imagery and the methods of Hu et al. (2013)

MODIS first must be processed from Level 1a MODIS data
(raw radiance counts) to Level 1b files (geolocated
at-aperture radiances) using NASA's SeaWiFS      Data    Analysis
System (SeaDAS).  These files were then atmospherically corrected
using the SeaDAS l2gen code, generating remote sensing reflectance,
chl-a, particulate backscattering       using   the      QAA
and normalized water-leavingradiance.

Inputs to the HAB MODIS model are:
1.Enhanced RGB (ERGB) image
2. Chl-a
3.Fluorescence Line Height (FLH)
4.Particulate   backscattering coefficient      ratio    at       551      nm
  calculated from: bbp,551/bbp,Morel*/

//--------Raster inputs from SeaDAS-------

//1. Enhanced RGB    (ERGB) image bands:
//Normalized water-leaving radiance (nLw) at: 547, 488 and 443 nm
//var MODIS_ERGB = ee.Image('GME/images/
14493593916934751216-11613121305523030954').select('b1', 'b2', 'b3');
var MODIS_ERGB = ee.Image('GME/images/
14493593916934751216-17488941626782682984').select('b1', 'b2', 'b3');

//2. Chl-a bands:
var MODIS_CHL = ee.Image('GME/images/
14493593916934751216-04594044960761723980').select('b1', 'b2')

//3. FLH bands (to be calculated)
var MODIS_FLH = ee.Image('GME/images/
14493593916934751216-11853667273131550346').select('b1','b2','b3')
```

```
//4.
//--a) particulate backscatterig coefficient at 551nm using QAA
var MODIS_bbp = ee.Image('GME/images/
14493593916934751216-08551536342581976716').select('b1')
//--b) bands for bbp Morel calculation
var MODIS_CHl = ee.Image('GME/images/
14493593916934751216-16143158689603361093').select('b1')

//-------Processing input rasters——

//-- Center and zoom to region of interest based on individual scene
//Scene based centering coded to allow application to any initial inputs
//Define scene extent of images
var SceneExtent = MODIS_CHL.select('b1');
var feature = ee.Feature(SceneExtent)
var GetBounds = feature.bounds()
//Get centroid of scene extent
var center = GetBounds.centroid(null, 'EPSG:4326')
var latlong = center.geometry()
//Convert centroid to numerical coorinates
var x= (latlong.coordinates().get(0).getInfo())
var y= (latlong.coordinates().get(1).getInfo())
var latlongnum= ee.List(latlong.coordinates())
print(latlongnum)
Map.setCenter(x,y,7)

//---Add land/water mask to entire image to allow better visualization of
//model input layers
var landwatermask = ee.Image('MODIS/MOD44W/
MOD44W_005_2000_02_24').select('water_mask');
print(landwatermask)
var vizParams = {
  min: 0,
  max: 1,
  palette: ['808080','000000'],
};
Map.addLayer(landwatermask, vizParams, 'Land water Mask')

//1. --- Visualize Enhanced RGB Image
var imageERGB = MODIS_ERGB.visualize({bands: ['b1', 'b2', 'b3'], max: 20});
Map.addLayer(imageERGB, null, 'ERGB')
```

```
//2. --- Calculate Chlorophyll from MODIS OC3 equation
//NASA OC3:
//Rrs1 = blue wavelength Rrs (e.g., 443, 490, or 510-nm)
//Rrs2 = green wavelength Rrs (e.g., 547, 555, or 565-nm)
//X = log10(Rrs1 / Rrs2)
//chlor_a = 10^(a0 + a1*X + a2*X^2 + a3*X^3 + a4*X^4)
var ReflRatio = MODIS_CHL.select('b1').divide(MODIS_CHL.select('b2'))
var logRefl = ReflRatio.log10()
var Chl_OC3 = MODIS_CHL.expression(
   '10**(0.2500 + -2.4752*R + 1.4061*(R**2) + -2.8233*(R**3) + 0.5405*(R**4))', {
     'R':logRefl
   })
var ChlViz = {min: 0, max: 10, palette: ['0000FF', '00FF00', 'FF0000']};
//Visualize CHL layer
Map.addLayer(Chl_OC3, ChlViz, 'CHL_A')

//3. --- Calculate FLH from respective loaded wavelengths
//First caluclate band ratio of respective bands
var BandRatio=ee.Number((746.3-676.7)/(746.3-665.1))
var FLH_baseline=
MODIS_FLH.select('b1').add((MODIS_FLH.select('b2').subtract(MODIS_FLH.select('b1'))).multipl
y(BandRatio))
//Calculate FLH for the entire scene
var FLH_scene = MODIS_FLH.select('b3').subtract(FLH_baseline);
//Visualize the FLH layer
var FLHViz = {min: 0, max: 1, palette: ['0000FF', '00FF00', 'FF0000']};
Map.addLayer(FLH_scene, FLHViz, 'FLH')


//4. --- Calculate bbp Morel from loaded SeaDAS bands
//First, mask out any Chl levels under 1.5 for calculation to exclude nonproductive waters
var ChlMask = MODIS_CHl.gt(1.5)
var Masked_chl = MODIS_CHl.mask(ChlMask)

//A. -- Add bbp from original SeaDAS file (bbp using Quasi-Analytical alogrithm)
var Masked_bbp = MODIS_bbp.mask(ChlMask)

//B. -- Calculate bbpMorel using the Morel (1988) algorithm
var bbpMorel = MODIS_CHL.expression(
   '(0.3*powChl)*(0.002+(0.02*(0.5-0.25*logChl)))', {
     'powChl': Masked_chl.pow(0.62),
     'logChl': Masked_chl.log10()
   })
```

```
//C. -- Calculate bbp ratio of bbpMorel/bbpQAA
var BbpRatio = Masked_bbp.divide(bbpMorel)

//Mask out values of zero (pixels without full ratio calculation)
var RatioMask = BbpRatio.gt(0)
var BbpRatioMasked = BbpRatio.mask(RatioMask)
//visualize bbpratio layer
var ratioViz = {min: .1, max: 2, palette: ['FF0000', '0000FF','008000' ]};
Map.addLayer(BbpRatioMasked, ratioViz, 'ratio')

//----Use given parameters to define classification regions of ocean color

//1. --- Define dark and bright ERGB areas for classification
//define the 25th percentile of reflectance for Red (448) and Green (557) bands

var ERGB_Red = imageERGB.select('vis-red').reduceRegion({
  reducer: ee.Reducer.percentile([25], ['25th'], null, null, null),
  geometry: feature.geometry(),
  scale: 30,
  maxPixels: 1e9
});
var ERGB_Green = imageERGB.select('vis-green').reduceRegion({
  reducer: ee.Reducer.percentile([25], ['25th'], null, null, null),
  geometry: feature.geometry(),
  scale: 30,
  maxPixels: 1e9
});

var Rednumber = ERGB_Red.get('vis-red');
var Greennumber= ERGB_Green.get('vis-green');

//Use the percentiles to extract the high and low reflectance of ERGB bands
var ERGBhighRefl = function (number_ith,Initial_scene) {
  var ERGB_ith_number = ee.Number(number_ith)
  var ERGB_ith_mask= Initial_scene.gt(ERGB_ith_number)
  var ERGB_ith_masked = Initial_scene.mask(ERGB_ith_mask)
  return ERGB_ith_masked
}
var ERGBlowRefl = function (number_ith,Initial_scene) {
  var ERGB_ith_number = ee.Number(number_ith)
  var ERGB_ith_mask= Initial_scene.lt(ERGB_ith_number)
  var ERGB_ith_masked = Initial_scene.mask(ERGB_ith_mask)
  return ERGB_ith_masked
}
```

```
//Apply to red (448) and green (557) bands
var RedPercentileupperRefl= ERGBhighRefl(Rednumber,(imageERGB.select('vis-red')))
var RedPercentilelowerRefl= ERGBlowRefl(Rednumber,(imageERGB.select('vis-red')))
var GreenPercentileupperRefl= ERGBhighRefl(Greennumber,(imageERGB.select('vis-
green')))
var GreenPercentilelowerRefl= ERGBlowRefl(Greennumber,(imageERGB.select('vis-
green')))

//Define bright and dark ERGB areas for final evaluation
var ERGB_bright=RedPercentileupperRefl.and(GreenPercentileupperRefl)
var ERGB_dark=RedPercentilelowerRefl.and(GreenPercentilelowerRefl)

//2. ---No additional processing of Chl scenes

//3. --- Define areas of high and low FLH values
//In order to find areas of high FLH, first define the 90th percentile of calculated FLH
var FLH_75th =FLH_scene.reduceRegion({
  reducer: ee.Reducer.percentile([90], ['75th'], null, null, null),
  geometry: feature.geometry(),
  scale: 30,
  maxPixels: 1e9
});
print(FLH_75th)
var number75th = FLH_75th.get('b3');


///Final.-----Defining categories of water and creating Chl. error

/*In order to combine water classificaitons into a single
values of each individual image must be set to a binary grid.
(ie. masked masked values must be converted into 0's in oder
for bands to be combined) */

var imagemask = function (Initial_image) {
  var imagemask=Initial_image.eq(1)
  var maskedimage=Initial_image.mask(imagemask)
  return maskedimage
}

var blank= ee.Image(0)
var Wherefunction= function (Initial_image){
  var output=blank.where(Initial_image.eq(1),1)
  return output
}
```

```
///---Compute chlorophyll error matrix (CDOM-rich water miss-classified
//as Chl-a from NASA OC3 algorithm)
var scaledFLH = FLH_scene.multiply(10)
var ChlError = MODIS_CHI.subtract(scaledFLH)
Map.addLayer(ChlError, null , 'Chlorophyll Error')

//----Give categorical definitions of each water classification
//based on previously defined parameters, and apply above
//functions to eliminate masked values
var SedimentRich =
Wherefunction(imagemask(ERGB_bright.and(FLH_scene.gt(ee.Number(number75th)))))
var ShallowClear =
Wherefunction(imagemask(ERGB_bright.and(FLH_scene.lt(ee.Number(number75th)))))
var CDOMRich =
Wherefunction(imagemask(ERGB_dark.and(FLH_scene.lt(ee.Number(number75th)))))
var PhytoRich =
Wherefunction(imagemask(ERGB_dark.and(FLH_scene.gt(ee.Number(number75th)))))
var HABPoor = Wherefunction(imagemask(PhytoRich.and(BbpRatio.gt(.8))))
var HABRich = Wherefunction(imagemask(PhytoRich.and(BbpRatio.lt(.8))));


//Merge all categories into a single 'classification' Image
//This image will have 5 bands, each with binary values
var mosaic=ee.Image.cat([SedimentRich,ShallowClear,CDOMRich,HABPoor,HABRich])
print(mosaic, 'mosaic')

//Covert binary layer into single band layer with values 1-5
//corresponding to class definitions
var CatImage = mosaic.expression(
  'Class1 + 2*Class2 + 3*Class3 + 4*Class4 + 5*Class5', {
    'Class1': mosaic.select('constant'),
    'Class2': mosaic.select('constant_1'),
    'Class3': mosaic.select('constant_2'),
    'Class4': mosaic.select('constant_3'),
    'Class5': mosaic.select('constant_4'),
});
var igppalette= {min: 0, max: 5, palette: ['FFFFFF','00FFFF','0000FF','00FF00', 'FF0000']}

//Mask out all zero values of final image to reveal Land/Wayer Mask
var CatImageMask = CatImage.gt(0)
var FinalClassified = CatImage.mask(CatImageMask)
//Visualize final classification image
Map.addLayer(FinalClassified, igppalette, 'Final Classified Water')
```