Birkett Water Mapping *NASA Goddard Innovation Lab*

Introduction

This project computes surface water extents from lake shapefile boundary maps. It uses the NRT Global Flood Mapping products produced from the LANCE-MODIS data processing system at NASA Goddard to compute the probability of water in each spatial cell within each shapefile boundary map. It then thresholds this probability to produce water mask files.

Workflow

This is a rough description of the processing workflow:

- Download MODIS floodmap data from https://floodmap.modaps.eosdis.nasa.gov/
  - Filename structure: MWP_2019244_120W040N_1D1OS.tif where "2019241" is Julian day of year, "120W040N" is location in geographic coordinates, and "1D1OS" refers to daily product
  - Each file represents 10 deg lat by 10 deg lon
- Make 8 day composites, where the first composite starts on day 1 of the year and includes days 1 – 8; second composite has days 9 – 16; etc.
  - Each file is an image classification with the following legend: 0 = bad data; 1 = land; 2 = permanent water; 3 = flood water.
  - Calculate "probability of water" as (# obs water/(# obs water + # obs land)).
  - Compute water masks by thresholding the water probabilities (threshold initially set to 0.5).
- Lake shapefiles exist for ~400 lakes (1 file per lake) that show the max extent of the lake plus a nominal buffer
  - Identify appropriate water mask tile to lake shapefile
  - Clip and reproject water mask to bounds of shapefile plus 2 km, use UTM local zone with target spatial resolution 250 x 250 meters
  - Calculate area of water within lake shapefile boundary
- Generate a textfile for each lake with columns containing information about the lake including area and rows are dates (relative to the 8 day composite)
  - Append area information to lake textfile.

Application Execution

These are the steps to install and run the application:

- Create the environment:

```
>> conda create --name floodmap
>> conda activate floodmap
(floodmap)>> conda install -c conda-forge xarray, numpy, rioxarray, shapely, regionmask, pandas, geopandas, wget
(floodmap)>> pip install wget
```

- Install the application from github:

```
(geoproc)>> git clone https://github.com/nasa-nccs-cds/floodmap.git
(geoproc)>> cd floodmap
(geoproc)>> python setup.py install
```

- Configure execution:

```
(geoproc)>> cp ./specs/sample-specs.yml ~/specs.yml
(geoproc)>> emacs ~/specs.yml
```

- Run the application:

```
(geoproc)>> python ./exe/floodmap.py ~/specs.yml
```

Application Configuration:

The application is configured using a yml file (called *specs.yml* in the example above). Here is a sample spec file (the parameters are explained in the next section):

```
defaults:
  results_dir: "/att/pubrepo/ILAB/projects/Birkett/MOD44W/results"
  year_range: [ 2014, 2019 ]
  day_range: [ 0, 360 ]
  water_class_thresholds: [ 0.02, 0.93 ]
  lake_masks:
    basedir: "/att/pubrepo/ILAB/projects/Birkett/MOD44W/lakes_lat_lon/"
    subdir: "{year}"
    file: "{lake_index}_{year}.tif"
    lake_index_range: [0,5000]
    mask: 3
    water: 1

  source:
   type: 'MPW'
   url:  'https://floodmap.modaps.eosdis.nasa.gov/Products'
   product:  '2D2OT'

  water_maps:
    threshold: 0.5
    bin_size: 8
```

Configuration Parameters

The application execution is configured using the following parameters in the spec.yml file:

results_dir: The directory where the processing results are written.

year_range: The range of years (inclusive) that will be processed.

day_range: The range of Julian days within each year that will be processed

water_class_thresholds: The min and max percent water thresholds calculated over time for each cell. Any cell with a percent water that falls below the min threshold is declared to be permanent land, and any cell with a percent water that falls above the max threshold is declared to be permanent water.

lake_masks: Specifications for reading the lake shapefile boundary maps.

basedir: Base directory of lake shapefiles.

subdir: Pattern describing subdirectory names.

file: Pattern describing file names

mask: Mask index value

water: Water index value.

lake_index_range: Range of lake indices that will be processed.

source: Specifications for reading the floodmap files.

type: Floodmap type: supported types= ['MPW']

url: URL of floodmap files.

product: Floodmap product (determines composite level).

water_maps: Parameters for the water mapping workflow:

threshold: Probability of water threshold for computing water masks.

bin_size: Number of days to composite when computing water masks.