

CZO Use Case - ArcGIS Extractors

For all ArcGIS extractors - Install ArcGIS Desktop Edition 10.3 with spatial analyst extension enabled. Also install arcpy and python that comes with ArcGIS. Set the PythonPath for the python under ArcGIS so that you can call arcpy anywhere in your system.

Extractors are in this repository:

<https://opensource.ncsa.illinois.edu/bitbucket/projects/CATS/repos/extractors-bd-cz/browse>

NDVI Extractor

Mosaics several scenes of Landsat 7 images, clips it based on a study region and computes Normalized Difference Vegetation Index (NDVI), and uploads the results to Clowder.

Input : A zip folder - earth-sundist.csv, zipped study region shapefile (Boundary.zip), two folders - Scene1 - (tiff files for band 3 and band 4, MTL file) and Scene2 (tiff files for band 3 and band 4, MTL file)

Output : A tiff file

Details provided by the student:-

1. This model currently is only works for Landsat 7. We need to change some parameters to make it work for Landsat 8.
2. For the solar angle, you can get it from metadata MTL file.

```
GROUP = IMAGE_ATTRIBUTES
CLOUD_COVER = 0.00
IMAGE_QUALITY = 9
SUN_AZIMUTH = 155.03337616
SUN_ELEVATION = 39.54882089
```

2. For the earth-sun distance, first you need to find date of year in the metadata MTL file:

```
GROUP = L1_METADATA_FILE
GROUP = METADATA_FILE_INFO
ORIGIN = "Image courtesy of the U.S. Geological Survey"
REQUEST_ID = "0101412026211_00196"
LANDSAT_SCENE_ID = "LE70230322000284EDC00"
FILE_DATE = 2014-12-03T03:44:34Z
STATION_ID = "EDC"
PROCESSING_SOFTWARE_VERSION = "LPGS_12.5.0"
DATA_CATEGORY = "NOMINAL"
```

3. Then you refer to the paper to get earth-sun distance from Table 6. (**Smruti:- I converted .xls file of this to csv and is in the zipped input sample file.**)



REU Summary Re...-Wei Quan.docx



REU Report.docx

Tree Delineation Extractor

Details provided by Kunxuan:-

Question (Smruti): Earlier I have sent you the working tree delination script (working_tree_delin_workflow_OSBS.py). While reviewing the script, Jong and I observed that lines (124 – 127) are independent of the rest of the above code. It takes two input files (v2014_OSBS_1_400000_3279000_dsm.tif and v2014_OSBS_1_400000_3279000_dtm.tif) and outputs (OSBS_01_chm.tif). As per the ppt that you have sent us earlier, the output of tree delination is a text file with tree centers and radii. In the line 120, the script outputs tree.csv file with this information. So, we are trying to understanding if lines (124-127) are required for the tree delination workflow?

As per your previous email description: "In the script, there are two different canopy height models (CHM). One is a product that came with the data, the other is calculated from the DSM and DTM which are extracted from the point cloud data using LASTools. The script currently use the first one, which is nicer, to perform tree extraction."

Is it the case that the input file (v2014_OSBS_1_400000_3279000_CHM.tif) and the file (OSBS_01_chm.tif) are the two different CHM that you were mentioning above? So, can we use either one as input to the tree delination workflow?

I have attached another script (tree_delin_workflow_2.py), where I am using (OSBS_01_chm.tif) as the input to the tree delination workflow.

The results are different in both cases.

Answer (Kunxuan):

This point was also discussed in the meeting we had a while back.

You are right in that the lines you mentioned are independent, and not needed for the script to work.

The two CHM files theoretically represent the same thing. The current script uses a CHM file that comes with the data as a processed data product.

For some data sets, this product might not exist. So Jong wanted me to include steps on deriving this product from data that every dataset should have, which is what lines 124-127 are doing. These steps should theoretically come before everything else to produce the CHM file that is used as input for the entire workflow.

The difference between the v2014_OSBS_1_400000_3279000_CHM.tif and OSBS_01_chm.tif is that the first is produced by the company who processes the data and they have their own algorithm and filters that makes everything "nice". The second is doing simple subtraction without any filters or smoothing, leading to less processed version of the same data. Therefore results should be different, but not by too much.

Rest of the materials are in the box folder - ppt, input and output files, and explanation document.



working_tree_de...orkflow_OSBS.py



tree_delin_workflow_2.py