Specklekiller

A SAR Backscatter Speckle Filter

Single dataset

## Usage

Packages required:

```
Required packages are included in the supplied environment filter.yml
```

## Input parameters:

```
# dir: input data set directory
# file: input data set (must be float32 ENVI 4 data type)
# wavdir: directory for wavelet data representation
# outdir: directory of the filtered data set
# outfile: filtered data set
# dp: number of pixel in the input data set
# dl: number of lines in the input data set
# skipdwt: switch set to 1 to skip the wavelet transform step
specklekiller(dir, file, wavdir, outdir, outfile, dp, dl, skipdwt)
```

## Input dataset

A single SAR image (float32 ENVI data type = 4)

The image must be a ground range detected or geocoded SAR backscatter image (amplitude or intensity).

## Sample call

```
From specklekiller import specklekiller
dir='G:/sentinelUruguay/envi/S1B_IW_GRDH_1SDV_20211212T092728_20211212T092753_029990_039495_987A_Cal'
file='Sigma0_VV_bswap'
dp=26009
dl=16864
wavdir='G:/sentinelUruguay/envi/S1B_IW_GRDH_1SDV_20211212T092728_20211212T092753_029990_039495_987A_Cal/wav'
outfile='Sigma0_VV_filt'
outdir=dir
specklekiller (dir, file, wavdir, outdir, outfile, dp, dl, 0)
```

## Test cases

The code was tested using one of the following SAR products (converted to an ENVI file):

- Sentinel-1 Ground Range Detected (GRD)

from Sentinels Scientific DataHub

https://scihub.copernicus.eu/dhus/#/home

Alaska Satellite Facility (https://vertex.daac.asf.alaska.edu/#)

- IceEye

(https://www.iceye.com/sar-data)

- TerraSAR-X/TanDEM-X

from Airbus (https://www.intelligence-airbusds.com/geostore/

- ALOS PALSAR

from Alaska Satellite Facility] (https://vertex.daac.asf.alaska.edu/

## Notes

Import from SNAP ENVI files:

```
SNAP exports amplitude and intensity products as ENVI files with data type 3 (long int 32 bit)
A data type 4 ENVI file compatible with the filter can be generated by procedure in module int32Tofloat.py:

int32Tofloat(dir, file, outdir, outfile, dp, dpout, dl)
where:
# dir: input data set directory
# file: input data set (SNAP ENVI 3 data type)
# outdir: directory of the coverted ENVI type 4 (float32) data set
# outfile: the coverted ENVI type 4 (float32) data set
# dp: number of pixel in the input data set
# dpout: number of pixels in the output data set (if data set needs to be resized)
# dl: number of lines in the input data set
```

Geocoded products from SNAP

SNAP exports geocoded products as σ^0 (dB) ENVI files (float32)
These files should be converted to intensity before filtering using:

```
sigmaToPow(dir, file, outdir, outfile, dp, dl)
```

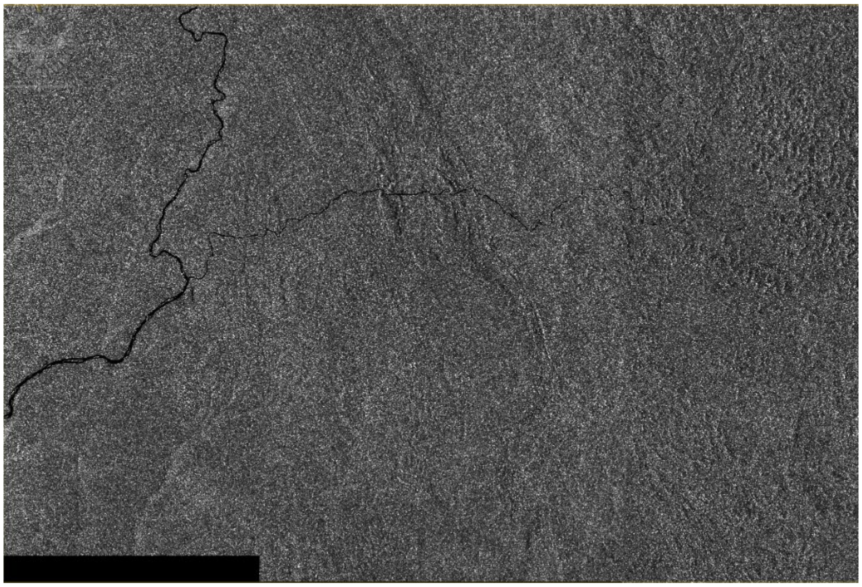
```
# dir: input data set directory
# file: input data set (SNAP sigma nought data)
# outdir: directory of the coverted intesity ENVI type 4 (float32) data
set
# outfile: the coverted intensity data set
# dp: number of pixel in the input data set
# dl: number of lines in the input data set
```

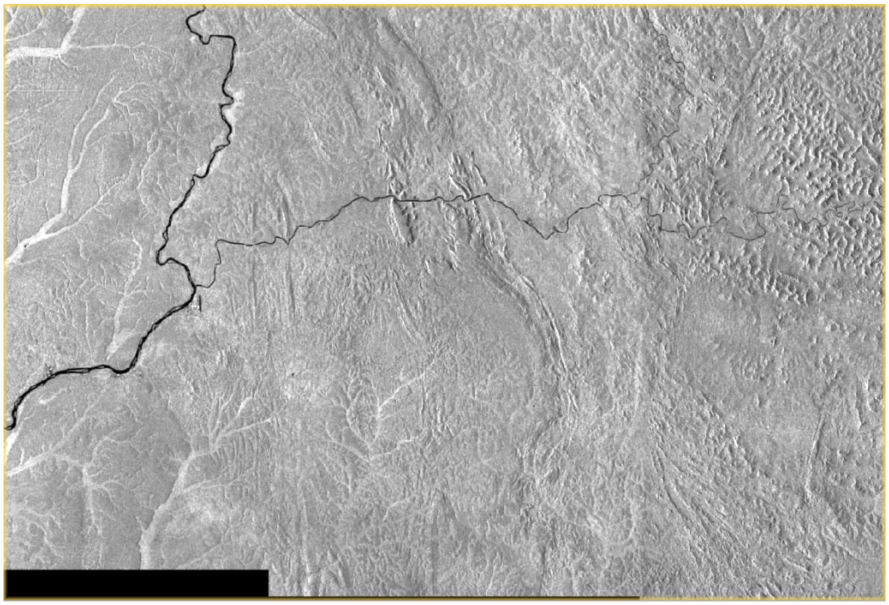
Import of ICEYE files

ICEYE files distributed as geofiff as exported by SNAP as uint16 with byte swapping
This data set must be converted to ENVI type 4 float32 by intTofloat.py:

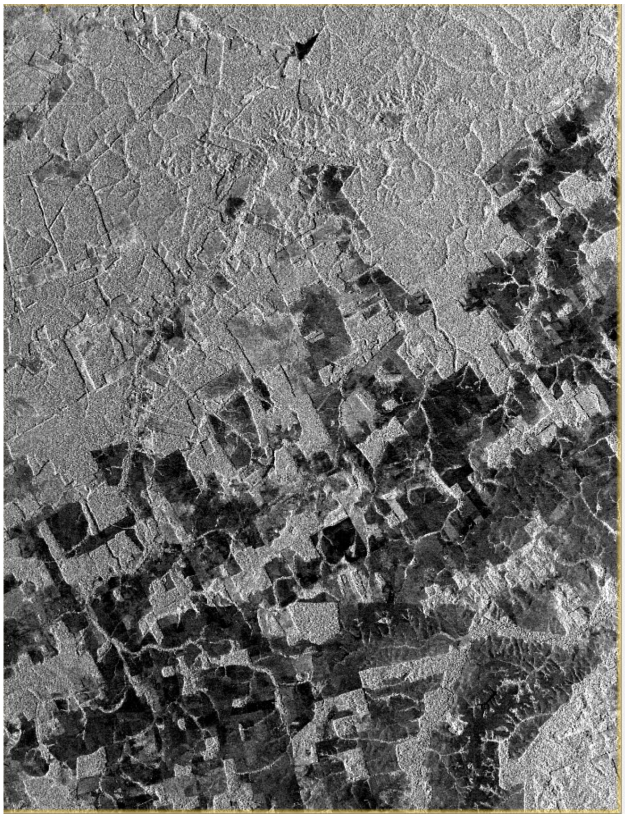
```
intoTofloat(dir, file, outdir, outfile, dp, dl)
where:
# dir: input data set directory
# file: input data set (SNAP ENVI 2 data type)
# outdir: directory of the coverted ENVI type 4 (float32) data set
# outfile: the coverted ENVI type 4 (float32) data set
# dp: number of pixel in the input data set
# dl: number of lines in the input data set
```

Examples – Sentinel-1 σ^0

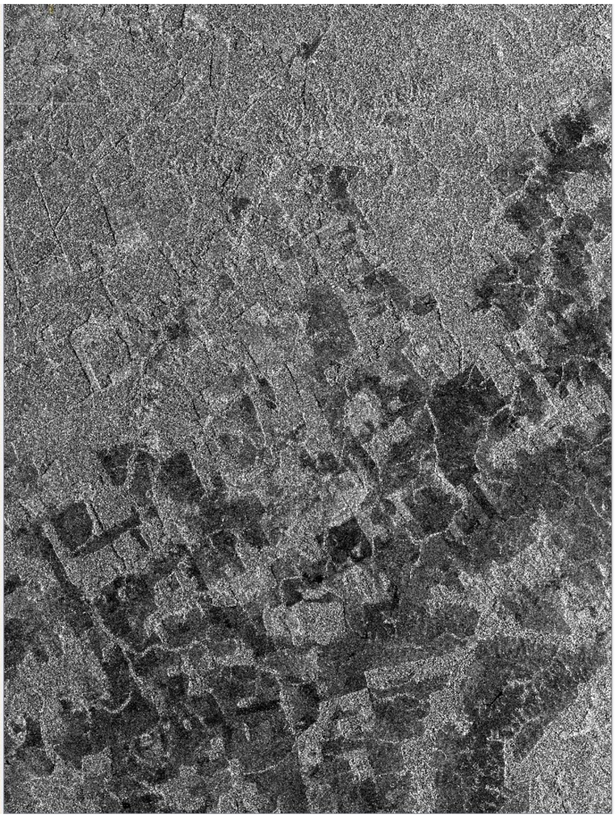




ICEYE Intensity data set



Specklekiller filtered set



SNAP filtered set