Radiometric Terrain Correction for Sentinel-1 Data
Sentinel-1 Toolbox software command-line recipe

This data recipe is for users who wish to radiometrically terrain correct (terrain-geocode) their Sentinel-1A data downloaded from ASF, using the Sentinel-1 toolbox (S1TBX) from the European Space Agency (ESA).

In this document you will find

1. **Background**
2. **Materials List**
3. **Steps**
4. **Output**
5. **How It Works**
6. **(Optional) View the RTC image in a GIS Program**
7. **Link to Sample Granule**
8. **Example Run**

1) Background:

Radiometric correction involves removing the misleading influence of topography on backscatter values. Terrain correction corrects geometric distortions that lead to geolocation errors. The distortions are induced by side-looking (rather than straight-down looking or nadir) imaging and are compounded by rugged terrain. Terrain correction moves image pixels into the proper spatial relationship with each other. Radiometric terrain correction combines both corrections to produce a superior product for science applications. This recipe is to support users who are comfortable working in the **command line** environment for post-processing SAR data. *Note: Windows users with insufficient memory may see error messages.*

ASF provides the python script "**procSentinelRTC_recipe.py**" to radiometrically terrain correct Sentinel-1A GRD data using the Sentinel-1 Toolbox software (S1TBX). This script uses a DEM file and a Sentinel-1A granule as inputs and creates terrain corrected geoTIFFs of each polarization found in the granule, an incidence angle map, and a DEM file.

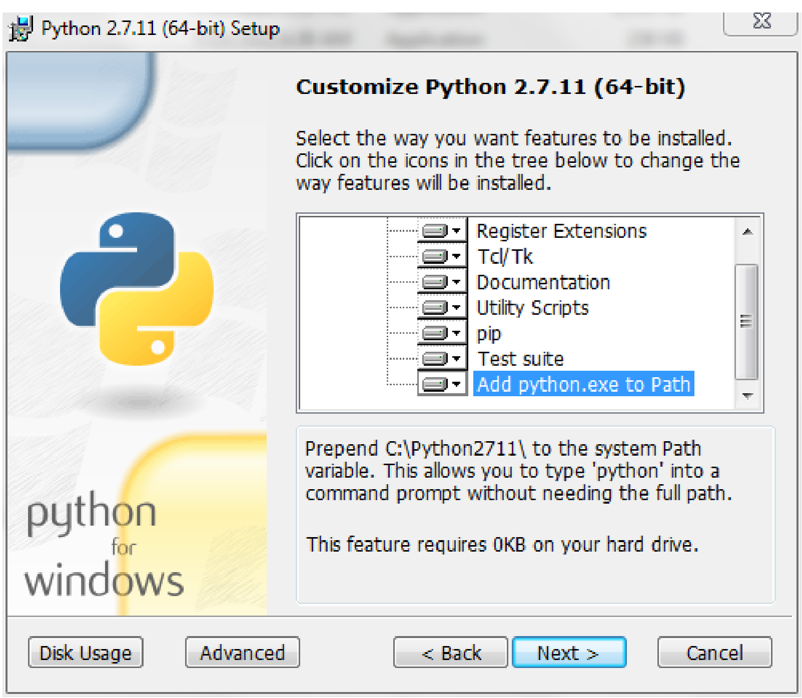
2) Materials:

- Sentinel-1A GRD product (download from Vertex)
- (Optional) Digital Elevation Model (DEM) (available from sources including

USGS Earth Explorer and OpenTopography; choose projection in meters)
- Sentinel-1 Toolbox
- Python™ software package
- OSGeo4W shell (part of any recent version of QGIS)
- procSentinelRTC_recipe.py (part of this package)

3) Steps:

1) Create a directory (e.g. S1TBX_processing_directory) to house the Sentinel-1A GRD products (the .zip file) and the *procSentinelRTC_recipe.py.* The S1A zip file must be in the directory where python script is run, or S1TBX will fail to process the granule further.
2) Download S1A GRD data from ASF Vertex and move it to your S1TBX processing directory.
   a. Start with our sample granule: Download Sample Granule
3) Download and install in your local environment the Sentinel-1 Toolbox package from ESA. **On Windows paths with spaces cause difficulties**, so the suggested location for the installation is "c:\s1tbx".
4) Download and install in your local environment the Python™ 2.7 software package.



For Windows users, it is essential to "Add python.exe to Path" (see on the left). A system administrator can help verify that the path is properly set in the environment variables for existing versions of Python.

5) Download procSentinelRTC_recipe.py (part of this package) into your S1TBX processing directory
6) (Optional) Download external DEM as GeoTIFF if desired (DEMs are available from USGS Earth Explorer and OpenTopography; choose projection in meters), and place in S1TBX processing directory. S1TBX will download and apply a DEM from the SRTM mission if no additional DEM is provided by you.
7) Download and install a recent version of QGIS, e.g. version 2.14 for the OSGeo4W shell.
8) Modify the directory path name to reflect your S1TBX processing directory in the script, procSentinelRTC_recipe.py. If you are a Mac user, set the value of baseSNAP to be the location of the S1TBX GPT executable, i.e. baseSNAP = '/Applications/snap/bin/gpt '
   If you are a Windows user, set the value of baseSNAP to the location of the S1TBX GPT executable, i.e. baseSNAP = "c:/snap/bin/gpt.exe "
9) For Windows users only: Start the OSGeo4W shell and change into the directory where the data recipe is stored.

10) (Optional) The following options are available when running the script:

-h, --help	show this help message and exit
-r PS	Pixel resolution - default = 10m
-d *DEM*, --dem=*DEM*	External *DEM file name*
-c, --clean	Clean intermediate files

11) Run script, “python procSentinelRTC_recipe.py [options] <S1A Zip File> <output directory>”

4) Output:

After the run is finished, there is a subdirectory named after the input data file as well as a “SAFE” directory containing the original data. Your data products are in the newly created file that matches your input file name. Your data products should be a GeoTIFF for each polarization of the SAR data (VV product or VV & VH products), a GeoTIFF of the DEM (DEM.tif) used for processing, and a GeoTIFF of the incidence angle map (INC.tif). All files are in the UTM projection.

```
$ cd TEMP

$ ls -1
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274.SAFE
```

A listing of the sub-directory shows:

```
$ ls -1
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL.data
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL.dim
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF.data
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF.dim
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC.data
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC_DEM.tif
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC.dim
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC_GVH.tif
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC_GVV.tif
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB_CAL_TF_TC_INC.tif
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB.data
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A
1_E274_OB.dim
```

The .data and .dim files are in native S1TBX format. The file extensions show the processing that was preformed:

- OB - Precise orbit applied
- CAL - Calibrated
- TF - Terrain Flattening
- TC - Terrain Correction

5) How it Works:

The procSentinelRTC_recipe.py script makes 4 calls to the S1TBX software gpt executable, followed by a conversion of the files from the ENVI format into GeoTIFFs. The steps are as follows:

1) Apply precise orbit information to the input file via automatic download
   gpt Apply-Orbit-File -PcontinueOnFail=true -PorbitType='Sentinel Precise (Auto Download)'

2) Radiometrically calibrate the image to Beta0 (the terrain flattening algorithm requires Beta0 input)
   gpt Calibration -PoutputBetaBand=true -PoutputSigmaBand=false

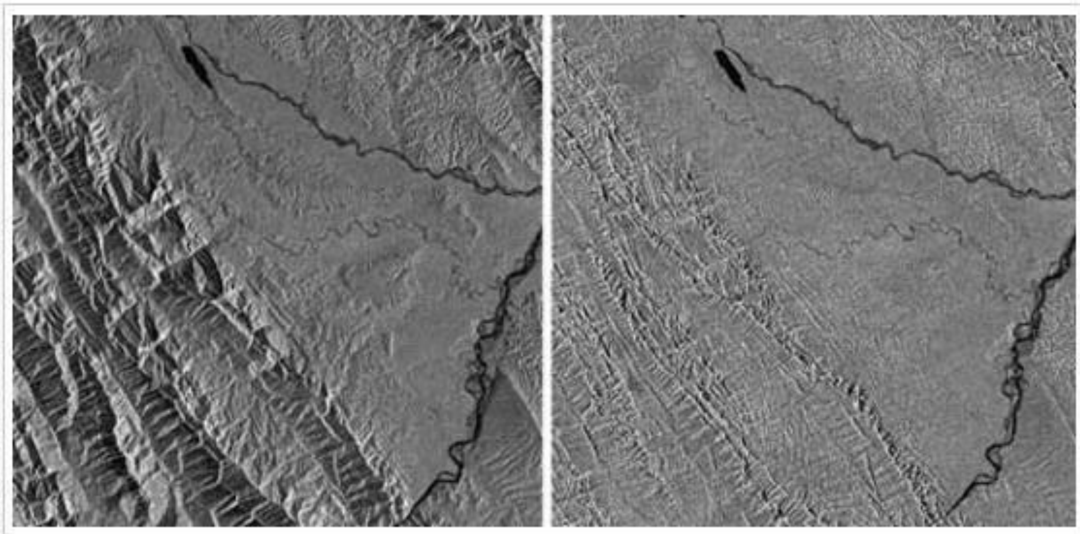
3) Radiometrically "flatten" the image, reducing terrain induced radiometric variations
   gpt Terrain-Flattening

4) Geometrically correct the image, reducing terrain induced distortions
   gpt Terrain-Correction -PsaveDEM=true -PsaveProjectedLocalIncidenceAngle=true \
   -PmapProjection=EPSG:<UTMCODE> -PpixelSpacingInMeter=<pixsiz>

6) (Optional) View the RTC image in a GIS Program

Note: The Gamma0 file type that the S1TBX RTC process produces is excellent for statistical comparisons, but can appear very dark. To view a lighter version of the image in a GIS environment, see the ASF data recipe ASF_DataRecipe_RTC_Viewing_Data_in_GIS.

7) [Link to Sample Granule]



Contains modified Copernicus Sentinel data (2015) processed by ESA

<u>Before and after</u>: In the uncorrected image (left), mountains in Bolivia appear to be stretched on one side and compressed on the other, and the properties of the land surface are misrepresented. In the RTC image (right), terrain correction has moved pixels, effectively sliding slopes into the correct geometry. Credit left image: Copernicus Sentinel data 2015. Credit right image: ASF DAAC 2016, contains modified Copernicus Sentinel data 2015.

Click above to download a sample, uncorrected Sentinel-1A granule. The image below uses a detail from the full granule.

8) Example Run:

$ procSentinelRTC_recipe.py -r 30
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274.zip
TEM
P
No external DEM file specified
INFILE =
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/S1A_IW_GRDH_1SDV_20150513T100637_201
50513T100702_005901_0079A1_E274.zip
BASENAME = S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274
TDIR = /hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE
DEM = None
CLEANUP = False
PIXEL SIZE = 30
Output directory: TEMP
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/TEMP/S1A_IW_GRDH_1SDV_20150513T10063
7_20150513T100702_005901_0079A1_E274

unzip
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/S1A_IW_GRDH_1SDV_20150513T100637_201
50513T100702_005901_0079A1_E274.zip
Found max_lon of -66.9374071361
Found min_lon of -69.6499055523
Found center_lon of -68.2936563442
Found UTM zone of 19
Found central meridian of -69
Found max_lat of -13.1862294574
Found min_lat of -15.2283333913
Found center_lat of -14.2072814244
Found hemisphere of S
Applying Precise Orbit file
/home/talogan/snap/bin/gpt Apply-Orbit-File -t
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/S1A_IW_GRDH_1SDV_20150513T100637_201
50513T100702_005901_0079A1_E274/S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005
901_0079A1_E274_OB-PcontinueOnFail="true"-PorbitType='Sentinel Precise(Auto Download)'
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/S1A_IW_GRDH_1SDV_20150513T100637_201
50513T100702_005901_0079A1_E274.zip
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274_OB.dim
Time to fix orbit: 150.0
ApplyingCalibration
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274_OB_CAL.dim
Time to calibrate: 81.0
Applying Terrain Flattening -- This will take some time
/home/talogan/snap/bin/gpt Terrain-Flattening -t
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/TEMP/S1A_IW_GRDH_1SDV_20150513T10063
7_20150513T100702_005901_0079A1_E274/S1A_IW_GRDH_1SDV_20150513T100637_20150513T10070
2_005901_0079A1_E274_OB_CAL_TF-
Ssource=/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/TEMP/S1A_IW_GRDH_1SDV_201505
13T100637_20150513T100702_005901_0079A1_E274/S1A_IW_GRDH_1SDV_20150513T100637_201505
13T100702_005901_0079A1_E274_OB_CAL.dim -PdemName="SRTM 1Sec HGT"
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274_OB_CAL_TF.dim
Time to terrain flatten: 870.0
Applying Terrain Correction -- This will take some time
/home/talogan/snap/bin/gpt Terrain-Correction -t
/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/TEMP/S1A_IW_GRDH_1SDV_20150513T10063
7_20150513T100702_005901_0079A1_E274/S1A_IW_GRDH_1SDV_20150513T100637_20150513T10070
2_005901_0079A1_E274_OB_CAL_TF_TC -
Ssource=/hugeslice/rtc/TOM/SENTINEL_RTC/JEREMY/SNAP/RECIPE/TEMP/S1A_IW_GRDH_1SDV_201505
13T100637_20150513T100702_005901_0079A1_E274/S1A_IW_GRDH_1SDV_20150513T100637_201505
13T100702_005901_0079A1_E274_OB_CAL_TF.dim -PsaveDEM=true -PsaveLocalIncidenceAngle=true -
PsaveProjectedLocalIncidenceAngle=true -PpixelSpacingInMeter=30 -PmapProjection=EPSG:32719 -
PdemName="SRTM1SecHGT"
S1A_IW_GRDH_1SDV_20150513T100637_20150513T100702_005901_0079A1_E274_OB_CAL_TF_TC.dim
Time to terrain correct: 272.0
Writing output file GVV
Writing output file GVH
Writing output file INC
Writing output file DEM
Time to export: 17.0
Total processing time: 1390.0