
Subject: Create ortho-rectified radiance image from HDF5 data

Posted by [Sean Hartling](#) on Tue, 10 Nov 2015 20:38:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

Does anyone have any advice on how to create an ortho-rectified ENVI data file that is geolocated from the lat/longs of HDF swath data?

Here is the code I am working with. This code produces a radiance image; however, it is not ortho-rectified according to the lat/lon from the HDF swath data, rather it creates the array from the tie point. I would like to tie the lat/lons from the original HDF5 data to each radiance pixel in the output. Any advice would be greatly appreciated.

```
PRO Hdf5_classic_VIS_projection
  compile_opt idl2
  ;
  ;e = ENVI(/HEADLESS)
  ; First restore all the base save files. This programs uses some ENVI routines
  envi, /restore_base_save_files
  ; Initialize ENVI and send all errors and warnings to the file batch.txt
  envi_batch_init, log_file='batch.txt'
  ;
  file_path = 'C:\Users\Sean\Desktop\Ball\'
  file = file_path + 'L1B_20150216_VIS.he5'
  ;
  ; Use a graphical user interface for viewing and reading HDF5 files.
  ; Result = H5_BROWSER(file)
  ; Define file dimentions
  exponent = intarr(1033, 99, 1072)
  Mantissa = intarr(1033, 99, 1072)
  L_lambda = fltarr(1033, 99, 1072)
  lat = fltarr(1033, 99)
  lon = fltarr(1033, 99)
  Solar_Zenith_Angle = fltarr(1033, 99)
  Sensor_refl = fltarr(1033, 99, 1072)
  surface_refl = fltarr(1033, 99, 1072)
  ; sample, row, spectral bands
  ;Dimensions: [1033, 99, 1072]
  ;
  ; Open the file
  file_id = H5F_OPEN(file)
  ; Accessing the datasets.
  ; This is located within '\SWATHS\').
  ; We could also have used H5G_OPEN to open up the group first.
  ;
  ; open, read and close radiance information
  dataset_id_exponent= H5D_OPEN(file_id,
```

```

'/HDFEOS/SWATHS/Radiance/DataFields/RadianceExponent')
exponent = H5D_Read(dataset_id_exponent) ; read teh file
H5D_CLOSE, dataset_id_exponent ; Close identifiers so we don't leak resources.
dataset_id_Mantissa= H5D_OPEN(file_id,
'/HDFEOS/SWATHS/Radiance/DataFields/RadianceMantissa')
Mantissa = H5D_Read(dataset_id_Mantissa)
; Open up the dataspace associated with the Mantissa image.
; dataspace_id = H5D_GET_SPACE(dataset_id_Mantissa)
H5D_CLOSE, dataset_id_Mantissa
; Calculate the radiance
Radiance = Mantissa*10.^(Exponent)
;
; open, read and close geolocation information
dataset_id_lat= H5D_OPEN(file_id, '/HDFEOS/SWATHS/Radiance/GeolocationFields/Latitude')
lat = H5D_Read(dataset_id_lat) ; read teh file
H5D_CLOSE, dataset_id_lat ; Close identifiers so we don't leak resources.
dataset_id_lon= H5D_OPEN(file_id,
'/HDFEOS/SWATHS/Radiance/GeolocationFields/Longitude')
lon = H5D_Read(dataset_id_lon)
H5D_CLOSE, dataset_id_lon
;
; open, read and close Earth Sun Distance (ESD) information
; just one value, 1.5155510e+011
dataset_id_ESD= H5D_OPEN(file_id,
'/HDFEOS/SWATHS/Radiance/GeolocationFields/EarthSunDistance' )
Earth_Sun_Distance = H5D_Read(dataset_id_ESD) ; read the distance file in meters
ESD = Earth_Sun_Distance*6.68458712*10.^(-12) ; should be in astronomical units
H5D_CLOSE, dataset_id_ESD ; Close identifiers so we don't leak resources.
; print, Earth_Sun_Distance
;
; read solar zenith angle (SZA)
dataset_id_SZA= H5D_OPEN(file_id,
'/HDFEOS/SWATHS/Radiance/GeolocationFields/SolarZenithAngle' )
Solar_Zenith_Angle = H5D_Read(dataset_id_SZA)
SZA = !pi*Solar_Zenith_Angle/180 ; convert degrees to radian
H5D_CLOSE, dataset_id_SZA
;
; Convert photon radiance Lq(nm) to spectral radiance L_lambda(um)
h = 6.626*10.^(-34) ; Plank's constant in Jeol per second J s
c = 2.99*10.^8 ; Speed of light in m/s
Wavelength_index = FINDGEN(1072) + 1.0
Wavelength = 0.279772*Wavelength_index + 404.68 - 3 ; wavelengths in VIS (nm)
;lambda = 550. ; Wavelength in nm
lambda = Wavelength*10.^(-9); Wavelength in meters
;Lq = 1.86*10.^13 ; extracted radiance value from GeoTASO image in Photons/(s cm^2 sr nm)
; conver the units of radiance to standard units
Lq = Radiance/(10.^(-4)*10.^(-3)) ; radiance in W/(m^2 sr um) <-- 1cm sq = 10.^(-4) sq m; nm =
10.^(-3) um

```

```

; read the solar constant for each wavelengths
ESUN=5000
; now calculate radiance
FOR k=0, n_elements(lambda)-1 DO BEGIN
    L_lambda[*,* ,k]= Lq[*,* ,k] * ((h*c)/lambda[k]) ; spectral radiance in W/(m^2 sr um)
    Sensor_refl[*,* ,k] = !pi*L_lambda[*,* ,k]*(ESD)^2/(ESUN*cos(SZA))
    ; surface_refl= Sensor_L_lambda[*,* ,k]/((ts*tv)+(1-sp))
ENDFOR
;
; use IDL routines to write the radiance file into a directory
Outputfile = 'C:\Users\Sean\Desktop\Ball\Out\Radiance_6.dat'
OpenW, Lun, Outputfile, /GET_LUN
WriteU,lun,L_lambda
Free_lun, Lun
;
; extract the lon/lat of the the first pixel
lo0 = lon[0] ; longitude of the first pixel
la0 = lat[0] ; latitude of the first pixel
;print, la0, lo0
;
units = envi_translate_projection_units('Degrees')
ps = [0.00007735D, 0.00246768D] ; Set the pixel size in degree (default)
mc = [0D, 0D, lo0, la0] ;and map tie point, -90.5597, 38.7053
map_info = envi_map_info_create(/geographic, mc=mc, ps=ps, units = units)

;create the header of file
ENVI_SETUP_HEAD, fname=outputfile, $
    ns=1033, nl=99, nb=1072, $
    interleave=0, data_type=4, $
    Sensor_type=Sensor_type, $
    map_info=map_info,$
    WAVELENGTH_UNIT=1, WL=Wavelength, $
    offset=0, /write, /open
; close the batch mode
;Envi_Batch_Exit
END

```
