Data Processing Overview and Protocol

This document was prepared based on my current data workflow NOTES files, R and Python scripts for the randomForest model estimation of population density based on a variety of covariates. This population density grid is then used as .

Overview of Protocol:

Pre-Processing Notes:

The data are downloaded from the NGA data archive and extracted using a script to process this data to calibrate and prepare it for use in FLAASH and subsequent analyses. I discovered that ENVI's preprocessing script is not capable of dealing with the L1T data and post-2004 conversion of measured at-sensor-radiance to scaled QCal DNs. Therefore I hardcoded the script to use Landsat 7 calculation methods incorporating a calculated gain and bias from the metadata LMax, LMin, QCalMax and QCalMin values. The resulting at-sensor-radiance values are correctly scaled to the stated LMax and LMin and should be used for further processing, noting that the radiance units are (W/(m2\*sr\*µm)).

NOTE: The above was true for ENVI 4.7, pre-SP2. It may have been fixed though I haven’t actually confirmed that this was the case. I prefer using my own script because I know exactly what it’s doing without relying on the default calibration black-box provided by ENVI.

The "\_asrad\_bil.dat" files in the folders here are suitable for processing in FLAASH after running through the "D:\Programming\IDL\Landsat\process\_l1t\_tmcal.pro" IDL script.

NOTE: One additional thing that may affect further processing, especially with FLAASH and its aerosol retrieval bands, is that band names in the "\_asrad\_bil.hdr" file should be edited down to something readable, e.g. remove the path for each band file added, etc. I suggest hand editing the ".hdr" file to rename the bands, as there seems to be a problem in the FLAASH interface for selecting bands with long names?

FLAASH Processing:

The FLAASH process was done after converting all of the Landsat data to asrad\_bil.dat files. The wavelengths and FWHM were imported from the text file in this directory, "Landsat Wavelengths and FWHM for FLAASH.txt" text file, using 1.0 scaling/multiply factors (wavelength units are in microns as expected). After clicing "Ok" on the import I set a "Use a single scale factor:" which needs to be adjusted from the default 1.0 to 10.0, because our radiance data are in (W/(m^2\*sr\*µm)) and to get it to (µW/(cm^2\*sr\*nm)) we have to divide by 10.0 (explained in the tutorial named below).

(µW/(cm^2\*sr\*nm)) == ((10^6\*µW/W)\*W/((10^4\*cm^2/m^2)\*m^2\*sr\*(10^3\*nm/µm)\*µm))

== (1/10)\*(W/(m^2\*sr\*µm))

The scene center and elevation information were pulled from the GCP data accompanying the L1T 2008 file for this GCP point:

Point\_ID 430280104  
 Latitude 45.864416  
 Longitude -117.767832  
 Height\_m 1481.878  
 Corresponding FLAASH setting:  
 Ground Elevation (km): 1.481878  
 Across\_Scan\_Residual\_m 4.336  
 Along\_Scan\_Residual\_m 2.689  
 ResidualY\_m 3.652  
 ResidualX\_m 3.543  
 Shape Point

The other relevant FLAASH settings used across all scenes (other than the flight date/time which was set based on the individual scene's metadata with the acquisition time being pulled from the GLOVIS metadata since the exact time is not included with the MTL file, only "DATEHOUR\_CONTACT\_PERIOD" is) (NOTE: as of 2011-03-09 this is not true, the scene center time is included with the MTL metadata) can be viewed in the FLAASH\_settings.txt file in each year's folder. One setting of note was that the "U.S. Standard" atmospheric model was chosen (since it's very dry, see the help file) and the "Rural" aerosol model was used with 2-Band K-T aerosol retrieval selected using Band 3 (upper) and Band 7 (lower) used for the ratio, as per the "Standard Over-land Retrieval" method. I also set the initial visibility to 80.00 km, since we can expect See the ENVI help file and recommendations in the FLAASH tutorial for ETM+ data:

D:\Documents\Graduate School\References\Remote Sensing\FLAASH Multispectral Tutorial.pdf

NOTE: The exception to this is for any image prior to 1986. There's a bug with the FLAASH user interface that does not allow you to set a year earlier than 1986. This date is used in the MODTRAN code to generate the estimated declination and right ascension of the sun. If you set an earlier date the interface nor FLAASH will complain, and proceed with processing, but it resets the year to 1986 (confirmed by looking at the saved "flaash\_template.txt" file in the journal folder). The declination and right ascension will most likely be different from year to year and so the correction for solar illumination will likely be off. Therefore, in order to get good results we need to choose a year after 1986 that has a declination and ascension that's close to the date of the image you are processing. I used the sunpos.pro IDL script downloaded from the IDL Astronomy Users' Library. You need to provide the date of the original image you're processing as in Julian form to the sunpos procedure. Usually a year that is a multiple of four after the image date you're processing will give declination/ascension that are similar and then you can adjust the day one day at a time to get close (within a hundredth or so of a degree).

So, for my 1984 image I did the following in IDL to find a date after 1986 where the declination and ascension values were similar enough for FLAASH/MODTRAN processing to be accurate:

jd = julday(10, 2, 1984, 18, 7, 36)  
 sunpos, jd, ra, dec  
 DEC -3.8506998586482015 Double   
 RA 188.92984744647046 Double

If we look at 1986, we can see that the DEC and RA are fairly different:

jd = julday(10, 2, 1986, 18, 7, 36)  
 sunpos, jd, ra, dec  
 DEC -3.6656963045316742 Double   
 RA 188.49600969427053 Double

So we're going to try the same date in 1988 to find an equivalent RA and DEC value:

jd = julday(10, 2, 1988, 18, 7, 36)  
 sunpos, jd, ra, dec  
 DEC -3.8654594771957074 Double   
 RA 188.96418619507810 Double

Since 1988 yields values fairly close to those of 1984, and it will be correctly processed by the FLAASH user interface and passed onto MODTRAN, we'll set the date for our 1984 FLAASH processing to be 1988-10-02 (see flaash\_template.txt in the 1984 folder).

NOTE: To open the resulting ENVI FLAASH files in ERDAS I needed to rename the layer names to remove the path in order to run the image through the Image Command tool and run the "Change Map Model" and "Change Map Project" in order to get it to open correctly with the right units.

Deshadowing/Topographic Normalization:

The DEM data were downloaded from the new National Map Seamless Request server with an AOI set out to just include Wallowa county. Derived hillshades were used for topographic normalization or "deshadowing":

1. Check the "D:\Documents\Graduate School\Research\Wallowa\Data\DEM\National Map, NED 10 Meter" folder for notes on processing the raw DEM data and for data acquisition in formation.

2. In order for most hillshading tools (including the one in ArcGIS) tool to accurately calculate the shading effects the file must have pixels in non-angular sizes, so at this point I reprojected the 4 scene mosaic from memory to a file in UTM, WGS84 Zone 11. I used a rigorous transformation and resampled the image to 10 m pixels with cubic convolution resampling algorithm.

3. I then stacked the first layer from the 2008 FLAASH-corrected Landsat data with the 10 m DEM and the subset mask data file with Exclusive overlap and cubic convolution resampling. This reduced the 10 m DEM to a 30 m pixel size for further processing. I unstacked and saved the single 30 m DEM layer to a new file, "DEM/National Map, NED 10 Meter/ned\_10m\_wgs84\_30m.dat" and exported that 30 m dataset to an IDL variable called "dem" for hillshading.

4. For each Landsat scene to be corrected for topographic illumination effects, I then calculated a hillshade image using the doshade.pro algorithm (Programming/IDL/Image Processing/), setting the azimuth and altitude of the sun from the values provided in the MTL header file for each Landsat scene.

2008:  
 SUN\_AZIMUTH = 142.1060726  
 SUN\_ELEVATION = 51.8039658

shadow = doshade(dem, [142.1060726, 51.8039658], 30, SHADING=hillshade)

2000:  
 SUN\_AZIMUTH = 133.1116427  
 SUN\_ELEVATION = 55.8787980

shadow = doshade(dem, [133.1116427, 55.8787980], 30, SHADING=hillshade)

1991:  
 SUN\_AZIMUTH = 130.1614026  
 SUN\_ELEVATION = 52.6576505

shadow = doshade(dem, [130.1614026, 52.6576505], 30, SHADING=hillshade)

1984:  
 SUN\_AZIMUTH = 150.9030937  
 SUN\_ELEVATION = 36.0143188

shadow = doshade(dem, [150.9030937, 36.0143188], 30, SHADING=hillshade)

5. The hillshade image is saved to the "hillshade" IDL variable. I imported this variable for each year back into ENVI. I then found a "flat spot" pixel by finding Wallowa lake and used the "Cursor Location Value" tool to note the illumination value of the the "flat spot" pixels. For example, in 2008 the "flat spot" pixel value is 0.785900. I then used band math to shift the illumination values up by a factor of (1 - "flat spot illumination"), so I ran the following band math over the hillshade image for each year:

2008:  
b1 + (1.0 - 0.785900)

2000:  
 b1 + (1.0 - 0.827853)

1991:  
 b1 + (1.0 - 0.795025)

1984:  
 b1 + (1.0 - 0.587987)

6. Finally, to correct the shifted hillshade image for largely spurious illumination values due to shadow/normal angle extremes, I used band math to truncate the values at 0.01 in order to reduce any extremely large values and sign-switching for reflectances:

(b1 ge 0.01) \* b1 + 0.01

7. The newly stretched data layer can then be divided through all of the layers of our FLAASH reflectances to correct and scale the FLAASH reflectance values to normalize for the effect of illumination intensity differences due to relief. The "flat spot" illumination should be divided by 1.0, while values with more illumination or less illumination are scaled towards that "flat spot" reference through the division. I stacked the FLAASH layers with the scaled relief layer and the "Data/Landsat/Subset/orcnty24\_wgs84\_wallowa\_prj\_subset\_mask.dat" file using exclusive extent and nearest neighbor resampling which will allow us to mask and subset the data at the same time that we topographically normalize. This data stack was then exported to IDL as a variable "data" and using the following IDL function topographic\_normalize, located in the "Data/Landsat/IDL and ENVI Expressions/topographic\_normalize.pro" file. I ran it using the following:

output = topographic\_normalize(data, data[\*,\*,6], subset=data[\*,\*,7])

The output variable can then be re-imported back into ENVI and the header information can be imported from the prior, non-normalized stack in order to preserve the map information and band names.

8. The normalized Landsat bands (just the FLAASH bands, not the hillshade or subset bands) were stacked using the ENVI layer stacking tool and saved into the "Deshadowed/L5043028\_JJJJYYYYMMDD\_vnmir\_flaash\_deshadow.dat" file in each year's folder.

9. The last step was to edit the header file to make the layer names more manageable.

RuleGen Classification:

From the original FLAASH file I calculated NDVI (defaults but set to Byte output instead of Floating) and TCA (which outputs as a signed 16-bit but that I stretched to signed 16-bit between 1-32,767). I did this by using the menu tools from the Transform menu in ENVI. These files I stacked together into a layerstack of 10 layers and then subset the layers using the subset EVF exported to an ROI from the Subset folder, setting values outside the county equal to -32,768.

Adding in the elevation-derived data to the layerstack, I included from the Oregon Geospatial Database dem10ne\_ files, the following:

dem10ne\_wgs84\_30m.img (DEM elevation values in feet)  
 dem10ne\_wgs84\_30m\_slope.img (Slope in degrees, byte scaled)  
 dem10ne\_wgs84\_30m\_hillshade\_stretched.img (Hillshade illumination created using ERDAS shaded relief with solar azimuth and elevation values pulled from the 2008 Landsat metadata, output as a float 0-1.0 (dem10ne\_wgs84\_30m\_hillshade.img) that I then scaled to 1-32,767). NOTE: I used 0 for the ambient illumination.  
 dem10ne\_wgs84\_30m\_aspect\_class.dat (A classified image, where I did some band math (see dem\_slope\_aspect\_classifying.exp) to calculate a three class image, 1 for pixels with a slope less than 5 degrees, 2 for pixels facing South, and 3 for North facing pixels)

For each year I created an ROI file with training polygons and pixels defined for the following classes (colors refer to a 5-4-3 composite):

1 Water  
 2 Bare - Snow, Ice, Sand  
 3 Bare - Rock, Built, Burned, Tilled  
 4 Vegetated – Riparian  
 5 Vegetated - Shrubby, Deforested  
 6 Grassland - Senesced, Tilled  
 7 Grassland - Productive, Non-irrigated Agriculture  
 8 Forested  
 9 Agriculture - Irrigated Hay (Bright Green)  
 10 Agriculture - Hayed (Pinkish)  
 11 Agriculture - Mature Wheat, Barley (Darkish)  
 12 Agriculture - Juvenile Wheat, Barley (Lightish Blue)

I defined ROIs that were non-changed land cover across all four years and added highly varialbe agriculture classes with a single pixel and representative colors, all saved to "all\_years\_vnmir\_flaash\_ndvi\_tca\_dem.roi" in the main Landsat folder. Using this as a starting point I further customized an individual ROI file for each year (located in the year folder) by including agricultural polygons (after removing the one pixel place holder).

Using these defined ROIs I ran the RuleGen classifier on the \_flaash\_ndvi\_tca.dat stacks. I set the RuleGen to defaults using the CRUISE algorithm, but changed the variable selection to be "2D" instead under the Advanced setup. (NOTE: For the 1984 image only 261 rules were generated, compared with over 400 or so for the other years. So I changed the variable selection to 2D and Prune Method to "Direct Stopping" instead of "Cross-validation" just for the 1984 image in order to create comparable results.)

Last, I built a mask for the subset area from the EVF file and applied it to the classified images, setting any values outside the subset area to 0.

Change Detection Analysis:

I combined classes into the following for further statistical analyses:

1 Bare, Built, Water (Class 3 from above)  
 2 Grassland or Agriculture (Class 4 from above)  
 3 Deciduous and Shrub (Class 5 from above)  
 4 Forest (Class 8 from above)

Fragstats Analysis:

I built Fragstats ready files by using band math to convert the 0 background value in the classified files to -99 to indicate background areas outside the region of interest by using the expression:

(b1 eq 0)\*(-99) + b1

Each of these classified files were saved into the folder:

D:\Documents\Graduate School\Research\Wallowa\Data\Landsat\Fragstats\Data\Land Cover\

I then recoded these files by property owner, with any areas outside the owner of interest area changed to negative. I created masks from the EVF files for each property owner type (having to reverse mask the Federal by Federal, Non-Wilderness/NRA mask to get just the wilderness/NRA area) by using the expression:

((b1 eq 0) and (b2 ne -99))\*(-1)\*b2 + ((b1 ne 0) or (b2 eq -99))\*b2

Where b1 is the property owner mask, 1 for inclusion in the owner's property, 0 for not) and the b2 is the recoded land cover data from above with -99 as the background value. These files were saved into the Land Cover by Owner folder in the data directory.

These files were listed in the Fragstats batch files and run using the saved Fragstats command files, with the output being saved and munged appropriately.