

HRSC data processing pipeline documentation

David Trethewey

December 13, 2016

Abstract

This is an attempt to document the data processing of the 179 High Resolution and Stereo Camera fields used in my 2014 MSc dissertation. The scripts processing the data were written on a somewhat ad-hoc basis at the time. To some extent the processing was documented in the dissertation itself, but this document will cover it at a more low level, as well as work done since the dissertation, such as the Top Trumps webpages.

1 Top-level summary of steps.

This is a list of the steps involved.

1. Download the [Colin Souness 2012 paper](#), with the supplementary data, including a list of the locations of the 1309 entries in his catalogue of Martian glacier-like forms.
2. Identify the HRSC coverage for the locations of the glaciers. This was done during the dissertation manually, using the web interface in the [Mars Orbital Data Explorer](#), to attempt to identify the HRSC tile that gave the best coverage for each Souness glacier. Later on, I made an automated check, using the HRSC DTM coverage shapefile available, along with shapefiles I had generated based on the Souness catalogue. This was using the OGR Python API.
3. Reproject the ND4 nadir image, and the DA4 areoid elevation to an equirectangular coordinate system, using 40° as the reference latitude. Done using the `gdalwarp` command line program, via scripting in Python (the `os.system` library). Resample a copy of the ND4 image to the resolution of the DTM.

4. Use [LandSerf](#) to create the derived topographic variables. Initially [GDAL](#) itself was used, but [LandSerf](#) was thought to offer more as regards the curvature layers. Used Landscript, which was generated by Python.
5. A 10 band layerstack including all the curvature layers was created for each tile, including [LandSerf](#)'s feature classification (1500m window), however in the further analysis a 6 band stack was used, including nadir, dtm, slope, aspect (from N), longitudinal curvature, and cross-sectional curvature. This was partly due to processing time, but also because including all of the curvature layers is duplicating the same information. Two different versions were used, using the absolute aspect from N, and the raw aspect (which has a discontinuity at 0/360°. The absolute aspect was used for the segmentation, but the raw for zonal stats (check detail of this).
6. Add 9999 to make the no data value zero, and avoid negative values of feature layers, which can cause problems with RSGISlib. Done with `gdal_calc`.
7. RSGISlib segmentation done - run via Python script for all of the tiles.
8. Populate stats to the raster attribute table of the segment clumps output of RSGISlib segmentation.
9. Export the RAT to a text .csv file.
10. Convert to shapefile with `gdal_polygonize`. Join stats to output shapefile. (check detail of this).
11. Zonal stats for each fields. Heads and extent areas of Souness glaciers.
12. Collect stats together for all fields.
13. Compare to overall distribution of variables in the HRSC tiles.
14. Gaussian fits to histograms, get functions of glacier-like-form abundance for each feature variable.
15. Create classifier functions.
16. Visualise in QGIS and do some basic statistical tests.
17. Subset the ND4 and DA4 images for Top Trumps. Colour the DA4 output with a pseudocolour scheme.

18. Generate JSON for Top Trumps.
19. Javascript for Top Trumps webpage.

2 Downloading data

Data is obtained from [Mars Orbital Data Explorer](#), using Data Product Search, selecting under the Mars Express heading ‘DTMRDR - Digital Terrain Map Reduced Data Record’. Originally when doing this manually, find footprints with ‘Find by Location or Feature’, specifying the latitude and longitude range. When filling in a few gaps where better coverage of a Souness GLF was found, I already know the product ID, so used ‘Filter by Product ID’.

Use the panchromatic nadir image and the areoid digital terrain model. These are the Product IDs that end in ‘ND4.IMG’ and ‘DA4.IMG’ respectively.

- 3 Reprojection and resampling
- 4 Creating derived topographic layers
- 5 Layerstacking
- 6 RSGISlib segmentation
- 7 Populating segment stats
- 8 Convert segments to shapefile, and zonal stats
- 9 Collating the stats
- 10 Creating the classifier function
- 11 Visualisation and statistical tests
- 12 Top Trumps