Integrated Software for Imagers and Spectrometers (ISIS) Version 3.10.2

Installation and Usage for Calibrating and Projecting Raw LROC NAC Images on Windows Systems

Le Corre, Daniel
Univeristy of Kent and ACRI-ST
Correspondence: dl387@kent.ac.uk

Introduction

The purpose of this document is to detail the process of installing and using the Integrated Software for Imagers and Spectrometers (ISIS) package for radiometric-calibration and map-projection of Lunar Reconnaissance Orbiter Camera (LROC) Narrow Angle Camera (NAC) images. The goal is to convert a raw LROC NAC PDS `.IMG` image file to a calibrated, echo-corrected and map-projected GeoTiff file ready for use in GIS software. Since the ISIS package is only available for Linux or macOS, installation onto Windows will make use of the Windows Subsystem for Linux (WSL) feature. WSL allows users to run a Linux distribution from their Windows system without the need for a virtual machine. Installation will also allow for the ISIS package to be run in the command line or as a graphical user interface (GUI). This document assumes that the necessary legacy base (as well as legacy LRO) ISIS data has been downloaded locally, or is able to be, before moving on to Section 2. The legacy base ISIS data should be installed along with kernels, although the LRO data does not need to be downloaded with kernels. Information on how to download the necessary ISIS data can be found on the US Geological Survey Astrogeology GitHub here. In order to batch-process several PDS `.IMG` files at once, Bash shell scripts have been provided which will loop through all such files and process them from `.IMG` to ISIS cube format and from ISIS cube to GeoTiff. These scripts can be downloaded from [LINK TO SHELL SCRIPTS].

Contents

1 Installation

This section will detail the necessary steps for installing ISIS version 3.10.2 for calibrating and map-projecting LROC NAC images.

1. Install Ubuntu WSL

   The first step is to install the Ubuntu WSL from the Microsoft store. The recommended version is Ubuntu 18.04.5 LTS. You will then be asked to create a username and password. Once completed, Ubuntu should then be updated to the latest version by running the following lines in the Ubuntu WSL terminal.

   ```
   $ sudo apt-get update
   $ sudo apt-get upgrade
   ```

2. Install VcXsrv Windows X Server and set up GUI

   Go to this link to download the installer for the VcXsrv X server for displaying the ISIS GUI. **NOTE:** When running the XLaunch program, make sure to disable access control.

   In order to set up the GUI, run the following lines in the Ubuntu WSL terminal. If the set-up has worked as expected, then a clock should appear on-screen after running `$ xclock`.

   ```
   $ sudo apt-get install -y -q xclip gnome-themes-standard gtk2-engines-murrine
   $ sudo apt-get install -y -q dbus dbus-x11 x11-apps libgl1-mesa-glx
   ```

```
$ echo export DISPLAY=:0 >>  /.bashrc
$ source  /.bashrc
$ xclock
```

3. Install Anaconda for Linux

   On Windows, download the latest Linux bash script installer for Linux here. The conda version that comes with Anaconda must be 4.8 or greater for the following installation to work. Then `cd` into the 'downloads' folder within Windows directory on the WSL which should be found within the WSL root directory at `/mnt/c/` for your C: drive. This will likely require read, write and/or execute permissions for your user to access files in the root directory. To get to the root directory run `$ sudo su`, then change the permissions by running `$ chmod u=rwx /mnt/`. Now running the following line will being the Anaconda install.

   ```
   $ bash Anaconda3-2022.10-Linux-x86_64.sh
   ```

4. Create conda environment and install ISIS package

   Create, activate and configure the conda environment in which the ISIS package will be installed by running:

   ```
   $ conda create -n isis
   $ conda activate isis
   $ conda config --env --add channels conda-forge
   ↳ --add channels usgs-astrogeology
   ```

   Ensure that the order of the channels is correct by running `$ conda config --show channels`. If the channel order is not `usgs-astrogeology`, `conda-forge`, then `defaults`, you must rerun the previous line but omit the `--add channels conda-forge` and check again. Once the channel order is correct, you can install the ISIS package, after which the conda environment must be reactivated.

   ```
   $ conda install -c usgs-astrogeology isis=3.10.2
   $ conda activate isis
   ```

   If the ISIS data has been installed locally to your Windows computer, perhaps to an external HDD due to the data's size, then run the following line (where path-to-folder is the path to where the data is stored on Windows) to set the environment variables `$ISISROOT` and `$ISIS3DATA`. **NOTE:** A $ immediately before some text denotes an environment variable (i.e. `$ISISROOT`, but a $ and a space before some text symbolises the command line.

   ```
   $ conda env config vars set ISISROOT=$CONDA_PREFIX
   ↳ ISIS3DATA=/mnt/path-to-folder
   ```

   Otherwise, you can run the following script which will automatically set the environment variables.

   ```
   $ python $CONDA_PREFIX/scripts/isis3VarInit.py
   ```

5. Test with an example LROC NAC image

   First `cd` into the `$ISIS3DATA` directory. Then use 'wget' to download the PDS `.IMG` file for the LROC NAC image "M1256183320RE" by running:

   ```
   $ wget -nd https://pds.lroc.asu.edu/data/LRO-L-LROC-2-EDR-V1.0/LROLRC_0032/
   ```

```
↳ DATA/ESM3/2017212/NAC/M1256183320RE.IMG
```

Next, convert the `.IMG` file to an ISIS cube by running the following.

```
$ lronac2isis from=M1256183320RE.IMG to=M1256183320RE.cub
```

Once completed, use `qview` to view the image as an ISIS cube in the GUI. You should see something which resembles Figure 1.1, in which case congratulations you have successfully installed ISIS on Windows and can move on to Section 2.

```
$ qview M1256183320RE.cub
```

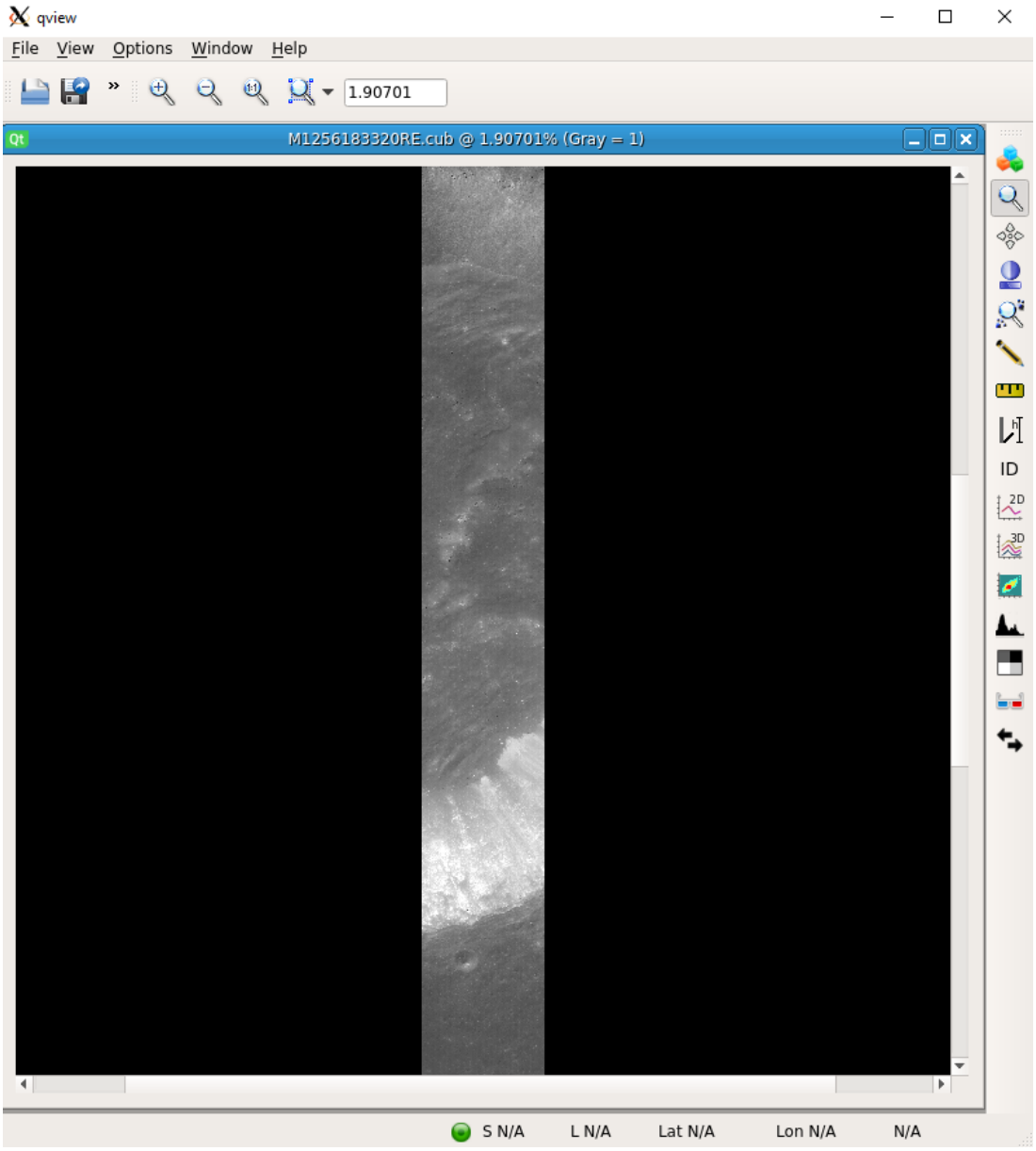

Figure 1.1: LROC NAC image "M1256183320RE" displayed within a GUI as an ISIS cube using the `qview` command.

2 Usage

Bash shell scripts have been provided to perform the various processing and file conversion for batches of files, as opposed to manually passing each command for all images. The first script is responsible for calibrating, echo-correcting and mapping the input raw `.IMG` files and converting them into ISIS

cube format. Whereas the other script should be called afterwards to convert these high-level ISIS cubes into GeoTiff format for use in GIS software such as QGIS [1] or JMARS [2].

2.1 Converting Shell Scripts to/from Unix

Since these scripts are intended to be run in a Linux terminal, there is the change that editing these files within Windows (if downloaded locally) will insert carriage return (CR) characters which are not used by Unix systems. If this is the case, the package `dos2unix` can be used to convert these files from Windows back to Unix and remove all CRs. `dos2unix` can be installed by running the following:

```
$ sudo apt-get install dos2unix
```

Next, in the Ubuntu WSL, `cd` into the folder where the bash shell scripts are located and run these lines in the terminal.

```
$ dos2unix LROC_NAC_process_all.sh
```

```
$ dos2unix LROC_NAC_convert_all.sh
```

2.2 Processing Raw PDS Image Products

In total, six ISIS commands are required for fully-processing a raw `.IMG` LROC NAC image and converting into a mapped ISIS cube. Each of these commands, along with their purpose are given below.

1. `lronac2isis` - converts an `.IMG` file to ISIS cube format
2. `spiceinit` - initialises SPICE kernels (web=true is set to use SPICE web service)
3. `lronaccal` - performs radiometric calibration on image
4. `lronacecho` - echo-corrects image (specific to LROC NAC)
5. `mosrange` - creates an individual `.map` file for each image
6. `cam2map` - map-projects image using `.map` file

To run the script for performing these processes (`LROC_NAC_process_all.sh`) the following line should be run in the command line from within the `$ISIS3DATA` directory. This script takes three optional arguments. The first is whether the intermediary files created by each of the above processes should or should not be deleted (a value of `1` or `0`, respectively - with the default being `1`). The second is whether the SPICE web service should or should not be used (`yes` or `no`, respectively - with a default of `no`). The final argument is the patch size that should be used when map-projecting the image. This should be the larger integer between (DEM/raw) or (output/raw) pixel resolutions. For example, for a 100 m/px DEM used to map a 0.5 m/px raw image to a 1 m/px output (i.e. $100/0.5 = 200$ and $1/0.5 = 2$), a patch size of 200 should be used. The default value is `50`.

```
$ cd $ISIS3DATA
```

`$ bash LROC_NAC_process_all.sh` (using the above default values)

`$ bash LROC_NAC_process_all.sh 0 yes 100` (for keeping files, using the SPICE web service, and a patch size of 100)

2.3 Mosaicking ISIS Cubes

These above processing steps are also performed by another script (`LROC_NAC_mosaic.sh`) which also performs photometric correction (using `photomet` and stitches the mapped cubes together to form a mosaic (using `automos`). This script takes the same three optional arguments as before, but also requires a name to save the mosaic under - typically representative of the region/landscape within it. The following line will process all raw `.IMG` files in a directory and create a photometrically corrected mosaic called '`moon_mosaic`'.

`$ bash LROC_NAC_mosaic.sh moon_mosaic` (using the default values)

`$ bash LROC_NAC_mosaic.sh moon_mosaic 0 yes 100` (for keeping files, using the SPICE web service, and a patch size of 100)

2.4 Translating from ISIS Cube to GeoTiff

For this conversion to work, we will need to install the GDAL package into its own conda environment, as was done for ISIS in Section 1. This can all be done by running the following lines from the base conda environment. If you are still in the ISIS conda environment then you can run `$ conda deactivate` which will return you to the base environment.

```
$ conda create -n gdal
$ conda activate gdal
$ conda install -c conda-forge gdal
```

Now that `gdal` has been installed, the bash shell script for converting from a mapped ISIS cube to a GeoTiff file (`LROC_NAC_convert_all.sh`) can be run. This will be looped for all such ISIS cube files in the `$ISIS3DATA` directory. The script will also convert to an 8-bit GeoTiff meaning that pixel values will range between 0 and 255. This script also takes the optional argument for deleting the intermediary files - in this case the mapped ISIS cubes. Therefore, to convert all mapped ISIS cube files to GeoTiff format, run the following lines.

```
$ cd $ISIS3DATA
$ bash LROC_NAC_convert_all.sh 0 (for keeping files)
```

3 References

[1] QGIS Development Team. QGIS Geographic Information System. QGIS Association, 2022. `https://www.qgis.org`.

[2] P. R. Christensen, E. Engle, S. Anwar, S. Dickenshied, D. Noss, N. Gorelick, and M. Weiss-Malik. JMARS - A Planetary GIS. In *AGU Fall Meeting Abstracts*, volume 2009, pages IN22A–06, December 2009.