

## 1 Files:

- pinhole.cc - the class that handles pinhole model cameras
- pinhole.h - the header file that defines the pinhole class
- CAHV.cc - the class that handles CAHV model cameras
- CAHV.h - the header file that defines the CAHV class
- isisInterfaceATK.cc - interface for ISIS camera models
- isisInterfaceATK.h - header file for ISIS camera models
- tests/make\_camera.cc - a basic tester that will create camera models
- tests/isistool.cc -

## 2 How to install:

### Pinhole and CAHV models

1. Install Prerequisites - Install Eigen 3.0, cmake
2. Install camera\_models - inside camera\_models/tests directory type: "cmake ."
3. Build camera\_models tests - inside camera\_models/tests directory type: "make"
4. Run Examples

### ISIS camera models

1. Get ISIS for your platform from

<http://isis.astrogeology.usgs.gov/documents/InstallGuide/>

You may pass to rsync the following options to decrease the download time: `-exclude '*/doc*' -exclude '*/*.png' -exclude '*/*.jpg' -exclude '*/*.html'`

2. For Linux only:

- Compile and install patchelf <http://releases.nixos.org/patchelf/patchelf-0.8/patchelf-0.8.tar.bz2>
- Get Qt 4.8:  
[http://download.qt-project.org/official\\_releases/qt/4.8/4.8.6/qt-everywhere-opensource-src-4.8.6.tar.gz](http://download.qt-project.org/official_releases/qt/4.8/4.8.6/qt-everywhere-opensource-src-4.8.6.tar.gz)  
Unpack and configure Qt. There is no need to compile it as we need only the headers (ISIS includes the libraries proper).

3. Install CSPICE: `wget ftp://naif.jpl.nasa.gov/pub/naif/toolkit/C/MacIntel_OSX_AppleC_64bit/packages/cspice.tar.Z` Unpack it. There is no need to compile it (as we need just the headers).  
To please ISIS, do  
`cd cspice`  
`ln -s include naif`

#### 4. Build using cmake:

```
cd lima/isis
mkdir build
cd build
cmake ..
```

If cmake fails to find the dependencies automatically, and the dependencies are installed in a certain directory (with standard include, lib, and bin subdirectories), you can pass that directory to cmake as

-DISIS\_ROOT:PATH=<your isis install dir>

If this still fails, you can set the paths to all dependencies individually, as follows.

On MacOSX:

```
cmake .. -DISIS_ROOT:PATH=<your isis install dir>
-DCSPICE_ROOT:PATH=<your cspice dir>
```

On Linux:

```
cmake .. -DISIS_ROOT:PATH=<your isis install dir>
-DCSPICE_ROOT:PATH=<your cspice dir>
-DPATCHELF:PATH=<path to your patchelf tool>
-DQT_ROOT:PATH=<path to your Qt installation>
```

Then type 'make'.

### 3 How to run:

1. ./make\_camera imageFilename configFilename resultsDirname
2. export ISISROOT=<your ISIS install dir>  
export ISIS3DATA=<your ISIS data dir>  
./isistool <your .cub file>

### 4 Camera configuration files:

#### 4.1 Pinhole

- [Required] CAMERA\_MATRIX f\_x 0 c\_x 0 f\_y c\_y 0 0 1
- [Required] WIDTH\_HEIGHT width height
- [Required] TRANSLATION t\_x t\_y t\_z
- [Required] ROTATION (0,0) (0,1) (0,2) (1,0) (1,1) (1,2) (2,0) (2,1) (2,2)
- [Optional] DISTORTION\_COEFFICIENTS

#### 4.2 CAHV

- [Required] C c\_0 c\_1 c\_2
- [Required] A a\_0 a\_1 a\_2

- [Required] H h\_0 h\_1 h\_2
- [Required] V v\_0 v\_1 v\_2
- [Required] WIDTH\_HEIGHT width height
- [Optional] QUATERNION q\_0 q\_1 q\_2 q\_3

## 5 Notes on Pinhole

Currently, the conversion from CAHV deviates from the paper in two places.

First in the equation for computing the principle point, equation 16 in the paper, the point is not being offset by  $\text{image\_width}/2$  or  $\text{image\_height}/2$ .  $H_c$  and  $V_c$  are being used directly. This is due to equation 16 producing incorrect results when using the offset. The offset may only be needed when converting to meters.

Second in the equation for extracting the rotation matrix from CAHV, equation 12 in ?, the paper calls for  $-V'$  and  $-A$ . Right now,  $V'$  and  $A$  are not being inverted. This is due to the conversion producing obviously wrong rotations when  $V'$  and  $A$  are inverted, but correct ones when they are not inverted.

These choices were made based on behavior when converting this image:

`image2surface_processing/tests/data/MLF_439225344RADLS0240000MCAM01870M1.IMG`

The paper includes a numeric example that can be used to verify conversions.