Calibrating and reading .DMCdata files to get pixel perfect output (not yet, but someday)
Michael Hirsch
Aug 2014

Given: you know UTC time range of interest
Want: data for that time range, registered to az/el per pixel
Assuming: you have the latest code from github.com/scienceopen in the directories
```
~/code/hist-utils
~/code/astrometry
```
as obtained by cd to those directories and typing
```
git pull
```
in each of those directories, or **if you don't yet** have those directories
```
cd ~/code
git clone https://github.com/scienceopen/hist-utils.git
git clone https://github.com/scienceopen/astrometry.git
```

Process:
(1) what frame indices do I need, given a UTC time range.
Example with 31 March 2014 Iridium 30 pass over HST0
```
cd ~/code/hist-utils
octave % feel free to use matlab, same results
format long g %to print number without exponent
fn = '/media/aurora1/HST2014image/2014-03-31/2014-03-31T06-12-CamSer7196.DMCdata';
utc(1) = datenum([2014 3 31 12 29 4]);
utc(2) = datenum([2014 3 31 12 29 24]); %assuming 20 second pass
utc2rawind(fn,utc)
```
% ans = 1197853 1198913 <-- estimate of frame number corresponding to 31 mar 2014 12:29:04 UTC

(2) make a calibration .fits file (if one hasn't already been made for this night, and no one has disturbed the camera physical mounting between the calibration and your data)

(2a) Let's choose to extract and average 10 frames just before satellite time
In Terminal:
```
cd ~/code/hist-utils
```

```
./rawDMCreader.py -i ~/Z/media/aurora1/HST2014image/2014-03-31/2014-03-31T06-12-CamSer7196.DMCdata -f
1197800 1197810 1 --fits --avg
```
using frames 1197800 to 1197810, this creates a fits file with the suffix _mean_frames.fits in the original .DMCdata directory

with a bit of circular logic, but let's check the frame time of this "mean frame", from matlab/octave:
```
datestr(getFrameUTC(fn,1197805),'yyyy-mm-ddTHH:MM:ss.fff')
```
ans = 2014-03-31T12:29:03.075
what we rely on here is that the star field apparent motion is vastly slower than the satellite motion through the field of view. This error source has not been formally analyzed.

(2b) now from Terminal, invoke astrometry.net program

```
cd ~/code/astrometry
```

```
./fits2azel.py -i /media/aurora1/HST2014image/2014-03-31/2014-03-31T06-12-CamSer7196_mean_frames.fits
-c 65.1186367 -147.432975 -t 2014-03-31T12:29:03.075 --mat
```

rename the calibration file

```
mv /media/aurora1/HST2014image/2014-03-31/2014-03-31T06-12-
CamSer7196_mean_frames.mat  /media/aurora1/HST2014image/2014-03-31/2014-03-31T06-
12-CamSer7196_azelcal.mat
```

This _azelcal.mat file can be used in your verifySatLocvsTime.m or wherever you need to have the best available estimate of az/el for each camera pixel.