Image calibration and reduction

Spring 2024

Make sure that the following data-crunching tasks are shared equitably among your team, as some of them will be repetitive and tedious. In most cases, the repetition and tedium can be automated, but at the cost of learning how to script using CCDStack's `Process Manager`, for which the learning curve is not necessarily very short.

Set aside a few uninterrupted hours remotely logged into the Astronomy Lab server, which has CCDStack installed. Locate the copies of your (or the archival) data on the disk. Make sure that there are copies of the current master dark and bias frames on the computer as well. Then start CCDStack.

CCDStack records the details of the data-file alteration in the FITS file headers, which will help you to keep track of what you have already done, either on purpose or inadvertently, to the data.

1 Processing calibration data

We take many bias and dark images in order that the signal-to-noise of our final calibrated images has not been degraded by the noise in the master bias and dark images. Because there are so many, we strongly encourage their reduction, and the creation of Master calibration images, on the TCS itself, and then transferring just the Master images to the Astronomy Lab server.

Most of the time, we will presume that the dark and bias data have already been processed but that, when possible, flat-field data will be taken every night and need processing. Do so in CCDStack with the `Process - Create Calibration Master - create master Flat` command. This will prompt you to select all the flat-field images in a given filter. You will also be prompted about bias and dark subtraction of the flat-field data; choose `Yes`, and choose the appropriate Master Bias frame.

In subsequent `Combine Settings` dialog, choose `sigma reject Mean` from the dropdown box; specify `3` and `1` for `sigma multiplier` and `iterations`; make sure the `normalize` box is checked; and click `OK`. In the next dialog, save the result under an evocative name, such as `Flat_Master_Red_mm-dd-yyyy`, in a location within easy reach of your other data files.

Repeat for each filter, giving in the end a separate flat field for each filter you used, to go with the master dark and bias frames.

2 Dark current and bias subtraction, and rejection of outlying hot and cold pixels

Dark current, bias, and responsive quantum efficiency vary from pixel to pixel rather than from place to place on the sky, so these corrections get made before the images are aligned. CCDStack's `Calibrate` routines scale the dark-current component of the master dark to the exposure time of each image and subtracts the properly-scaled dark and bias all with one command.

Large-format detector arrays always have some pixels with large and variable dark current, and some dead pixels that draw no current at all. These "hot" and "cold" pixels need to be identified by comparison of the frames at the same wavelength and target, as a given pixel of these types will be problematic in every image. They can be corrected by interpolation among neighboring pixels. That is what the `Data Reject` routines do; they have only one threshold setting, though, so hot and cold pixels need to be corrected in separate operations.

Dark current and bias do not depend on wavelength, but flat-field does. Thus, the calibration and hot/cold corrections are performed on one filter and target at a time, using these CCDStack commands:

1. `File` — `Open Selected`, using the path and file strings to identify data for a given filter
2. `Process` — `Calibrate`, using the current master dark, bias, and flat frames (if available)
3. `Process` — `Data Reject` — `Procedures`
   (a) `reject hot pixels` with a strength of 30, then `Apply To All`
   (b) `interpolate rejected pixels` with 10 iterations, then `Apply To All`
   (c) Repeat steps a–b for the cold pixels.
4. `File` — `Save` — `All`, choosing some descriptive file suffix (e.g. `_Cal`)
5. `File` — `Remove all images - Clear`

3 Alignment

Before they are normalized, all the frames need to be shifted so that each star occupies the same pixels in each frame. This is what `Stack - Register` commands are for. When starting these commands, CCDStack will take the image currently displayed and automatically select a set of stars from this image to use as alignment references. Examine the stack of images to see if the alignment stars seem well chosen; clear and select different ones if not. (I always choose my own, and I usually choose at least twice as many as CCDStack does on its own.) For images in which the stars are more than a fraction of stellar image size off the mark, click and drag them into better alignment with the mouse before pushing the `align all` button. `Star Snap` does a coarse alignment, and the `Bicubic B-Spline` fit does a very precise alignment. Here is the sequence of commands:

1. `File` — `Open Selected`, using the path and file strings to identify `_CAL` data for a given project.
2. `Stack` — `Register`
   (a) `Star Snap` — `remove all reference stars` — `select/remove reference stars`, following the prompts, then `align all`
   (b) `Apply` — `Nearest Neighbor` — `Apply To All`
3. `File` — `Save` — `All`, again choosing some descriptive file suffix (e.g. `_ALIGN`)
4. `File` — `Remove all images - Clear`

Three related items worth noting:

- It is probably best to align the frames one target and filter at a time. But sometimes it works, and saves a little time, to align all the filters with the same binning simultaneously.
- When running within `Process Manager`, click the `Pause` box to interrupt the script. The rough alignment before `Star Snap` has to be done by hand by dragging the stars into alignment, because…
- The same `Stack - Register` routines can be used to **tessellate** images: align images with large offsets, on the way to making a mosaic of camera fields.

4 Normalization, removal of cosmic-ray hits and satellite trails, and averaging

These steps also need to be done one target and filter at a time. In each case, choose the frame taken with the largest elevation angle, to which to normalize the others. If running within `Process Manager`, check the `Pause` box to interrupt the script; fields that represent the image highlights and the background need to be identified by hand.

This is where artifacts peculiar to individual frames are removed. Cosmic-ray hits and satellite trails are different from frame to frame, so a search of the stack of frames to find things that only appear in one can allow identification and removal by interpolation. As with hot and cold pixels, this is a job for `Stack - Data Reject`, but this time it is best to reject a pixel according to deviation from the average of the same pixel in the stack's images.

1. `File` — `Open Selected`, using the path and file strings to identify `_CAL_ALIGN` data for a given object.
2. Navigate through the stack to display the image taken closest to transit, or to an A0V star observation. Use arrows at the top of the toolbar on the left of the CCDStack window to do this.
3. `Stack` — `Normalize` — `Control` — `Both`, selecting areas of "blank" sky and highlights of the targets in response to the dialog boxes.
4. `Stack` — `Data Reject` — `Procedures`
   (a) `Poisson sigma reject` using a value of 2.2 for the sigma multiplier, then `Apply to All`
   (b) `interpolate rejected pixels` — `Apply to All`
5. `File` — `Save` — `All`, again choosing some descriptive file suffix (e.g. `_NORM`)
6. `Stack` — `Combine` — `Mean`
7. `File` — `Save` — `This`, accepting the prepended `Mean` on the file name
8. `File` — `Remove all images - Clear`

5 Align the average images

If you are combining multiple nights of data, you will want to create an average image of all of the nightly average images that you have available. Open all of the images that you want to average together, and page through them to check that there is no rotation offset between them. If there is, then use `https://nova.astrometry.net/` to determine the rotation between the frames. Use CCDStack's Transform tool (`Edit - Transform - Rotate - Specify Rotation Angle`) to rotate each image as needed. Then, start the stack alignment and average process:

1. `Stack - Register - Star Snap - align all`
2. `Stack - Register - Apply - Bicubic B-Spline - Apply to All`
3. `Stack - Combine - Mean`
4. `File - Save - This`, accepting the prepended `Mean` on the file name
5. `File - Remove all images - Clear`

Repeat this for each filter.

Presumably, the L image was taken in 1×1 binning, and the others 2×2. Open the L image first, then the others; this automatically interpolates the 2×2 images to 1×1. Then display the L image, and start the stack alignment process:

1. `Stack - Register - Star Snap - align all`
2. `Stack - Register - Apply - Bicubic B-Spline - Apply to All`
3. `File - Save Data - All`, again choosing some descriptive file suffix (e.g. `_REALIGN`)
4. `File - Clear`

As usual, it will be necessary to examine the choice of alignment stars, and to drag each of the monochrome images into alignment with L, before the `Star Snap` step.

Now you have calibrated and aligned images of the target in all filters. You can proceed to do photometry confident that the stars have the same coordinates in each image, and you can begin color composition of potentially pretty pictures of your target.