Image calibration and reduction

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 away, 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 with one of the astronomy-lab computers which has CCDStack installed. Copy your data from the Wiki onto the disk. Make sure there are copies of the current master dark and bias frames on the computer as well. Then start CCDStack.

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

1. Processing calibration data

We can presume most of the time that the dark and bias data have already been processed, but that 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 and Dark frames.

In the 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 four separate flat fields to go with the master dark and bias frames.

2. Dark current and bias subtraction, flat fielding, 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; subtracts the properly-scaled dark and bias; and divides by the master flat field, 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's what the Data Reject routines do; they only have one threshold setting though, so hot and cold pixels need to be corrected in separate operations.

Dark current and bias don't depend on wavelength, but flat-field does. Thus one calibrates and hot/cold corrects one filter and target at a time, using these CCDStack commands:

File > Open Selected, using the path and file strings to identify data for a given filter
Process > Calibrate, using your new flats, along with the current master dark and bias frames
Process > Data Reject > reject hot pixels > Apply To All
Process > Data Reject > interpolate rejected pixels >Apply To All
Process > Data Reject > reject cold pixels >Apply To All

Process > Data Reject > interpolate rejected pixels >Apply To All
File > Save Data > All, choosing some evocative file suffix (e.g. .CAL).
File > Clear

3. Alignment

Before they are averaged, all the frames need to be shifted so each star occupies the same pixels in each frame. This what the 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 a 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's the sequence of commands:

File > Open Selected, using the path and file strings to identify .CAL data for a given object
Stack > Register > Star Snap > align all
Stack > Register > Apply > Bicubic B-Spline > Apply to All
File > Save Data > All, again choosing some evocative file suffix (e.g. .ALIGN)
File > Clear

Three related items worth noting:

- It is probably best to do align 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, check 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 **tesselate** images: that is, to 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's best to reject a pixel according to deviation from the average of the same pixel in the stack's images.

File > Open Selected, using the path and file strings to identify .CAL.ALIGN data for a given object

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.
Stack > Normalize > Control > Both, selecting areas of "blank" sky and highlights of the targets in response to the dialog boxes
Stack > Data Reject > Procedures > Poisson sigma reject > Apply to All, using a value of 2.2 for the sigma multiplier
Stack > Data Reject > Procedures > interpolate rejected pixels > Apply to All
File > Save Data > All, again choosing some evocative file suffix (e.g. .NORM)
Stack > Combine > Mean
File > Save Data > This, accepting the prepended Mean on the file name
File > Clear

5. Align the average images

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:

Stack > Register > Star Snap > align all
Stack > Register > Apply > Bicubic B-Spline > Apply to All
File > Save Data > All, again choosing some evocative file suffix (e.g. .REALIGN)
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 the 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.