Image reduction and analysis at the Peter van de Kamp Observatory

Software

The software we use to reduce (apply calibrations to) data is *AstroImageJ*. We also use it to make images and to analyze photometric variability (measure a star’s relative brightness over the course of many observations) of stars that may host transiting exoplanets.

The software is freely available:

[www.astro.louisville.edu/software/astroimagej/](http://www.astro.louisville.edu/software/astroimagej/)

It runs on Macs, PCs, and under Linux.

Data Reduction

The basic unit of observing is a single image. Color images of extended objects (like planets, nebulae, and galaxies) are made by combining separate images each taken with a different color filter. And exoplanet transit light curves are made from a sequence of images that contains the host star (not necessarily taken in more than one filter).

Images are produced by our CCD detector: A Charge-Coupled Device (CCD) is the common type of electronic light sensor in nearly all cameras these days. Astronomical CCDs are bigger, have smaller pixels, and have better (lower) noise properties than standard commercial CCDs. When light hits a CCD it induces a charge in the pixel in which it lands. The raw data of an image is a map of the charge in each pixel. You can forget the physics and think of the image as an array of integers with values proportional to the amount of light that landed on that pixel during the time the CCD was exposed to light. But the raw data also contain noise from the detector itself. Before we analyze our data or make our images, we need to remove as much of this noise as possible.

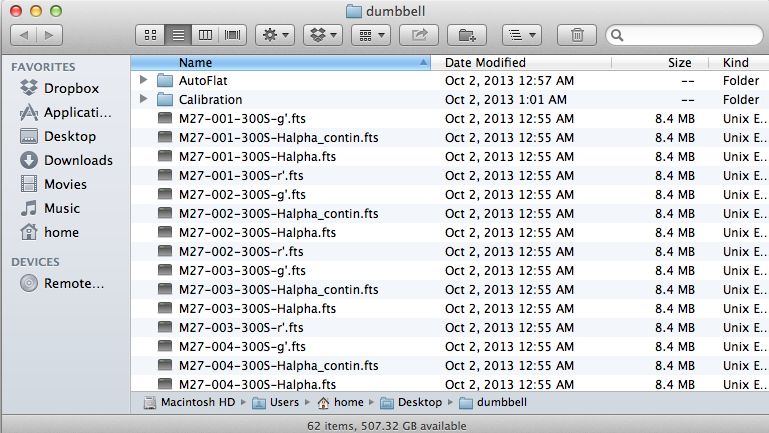
The reduction of most observations is similar: three types of corrections are made to each image: a bias correction removes the more or less constant signal caused by the voltage applied to the CCD – a sort of “zero point” – that you get even without exposing the CCD at all; a dark current correction which accounts for the thermal noise (charges induced in pixels by heat rather than light; we cool our CCD typically to -20 degrees C to minimize thermal noise, but you can’t eliminate it); and a flat-field correction that corrects for pixel-to-pixel sensitivity variations (so that a uniformly bright source of light would induce the same signal in each pixel). Each of these corrections is made by empirically measuring the thing you’re trying to calibrate (the bias level, the thermal noise, and an image of a uniform light source) and then subtracting or dividing out that noise estimate.

The bias is measured by taking many zero-second long exposures, and then averaging the resulting images. The dark current is measured by taking many exposures that are the same length as the data observations, and averaging them together. And the flat field is measured by observing a patch of the dusk sky (“sky flats”), and averaging them together.

To get an idea of what images taken with the telescope and CCD camera look like, download the archive of images of Messier 27 (M27), also known as the Dumbbell Nebula.

<http://astro.swarthmore.edu/~cohen/telescope/dumbbell.zip>

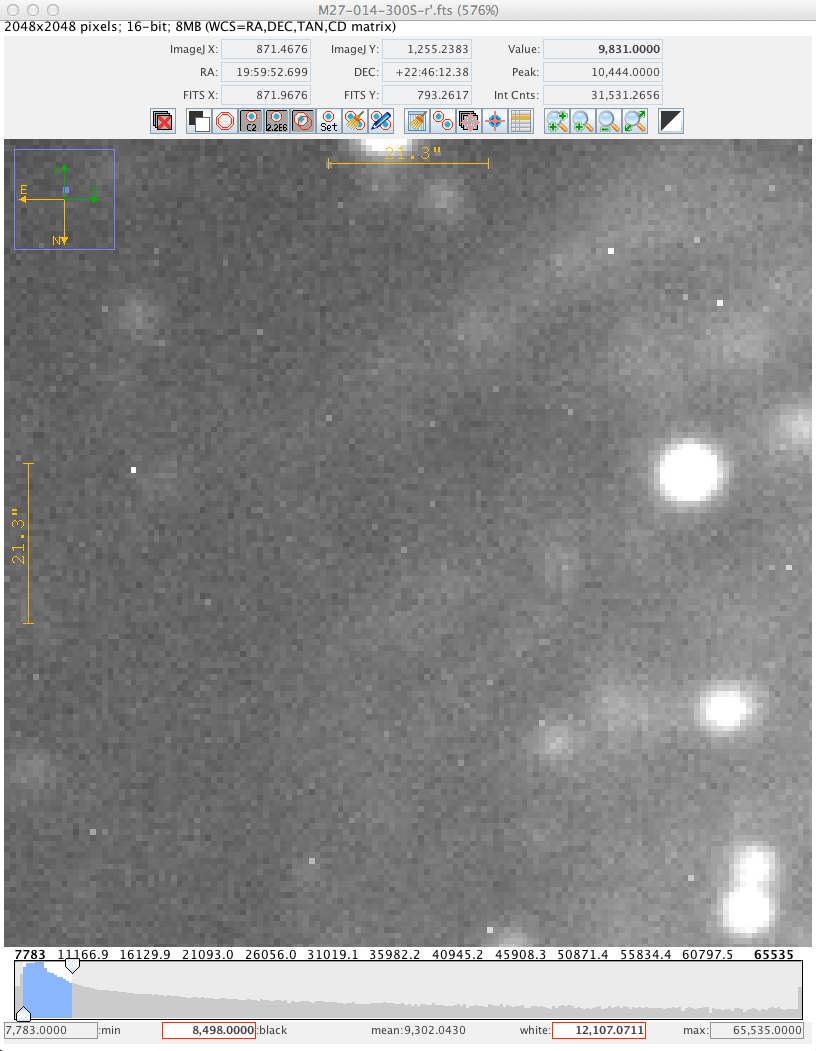
When you unpack the archive, you’ll have sixty images (15 in each of four filters) and two folders (Calibration and AutoFlat). The AutoFlat folder has several images using each of the four filters (indicated in the file names, just like the data images are). The Calibration folder has 31 bias images and darks with various exposure times (indicated in the file names).



Explore a bit before you reduce the data. Open up a representative source image in *AstroImageJ*. First, note that there are 15 300-second-long observations in each of four filters (so, sixty image files in FITS format (.fts), which can be read by *AstroImageJ*). Use the normal *file* > *open* menu to open one of the g-filter images. Then do the same for r-filter, H-alpha, and H-alpha-continuum (all from the same number observation, ideally). Note that as you mouse over an image, some numbers at the top change. The "Value" is the number of counts in the pixel the cursor is over; it is what determines the shade of gray that’s shown in that pixel (whiter means higher numbers).

Note also that with an image in the foreground, the *edit* > *FITS header* menu shows you the so-called header (full of descriptive information - metadata! - about the image; see, e.g., the exposure time (300 seconds) on line 9).

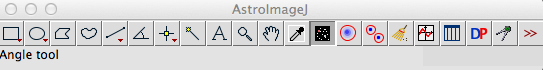
Try changing the grayscale in one of your open images using the horizontal sliders near the bottom of the window. Zoom in on a star in. See how many pixels are in any one star’s image. Note that the only thing that gives the image of the star perceptible size is the blurring effect of the Earth’s atmosphere. The true size of any star is much, much smaller than a single pixel.



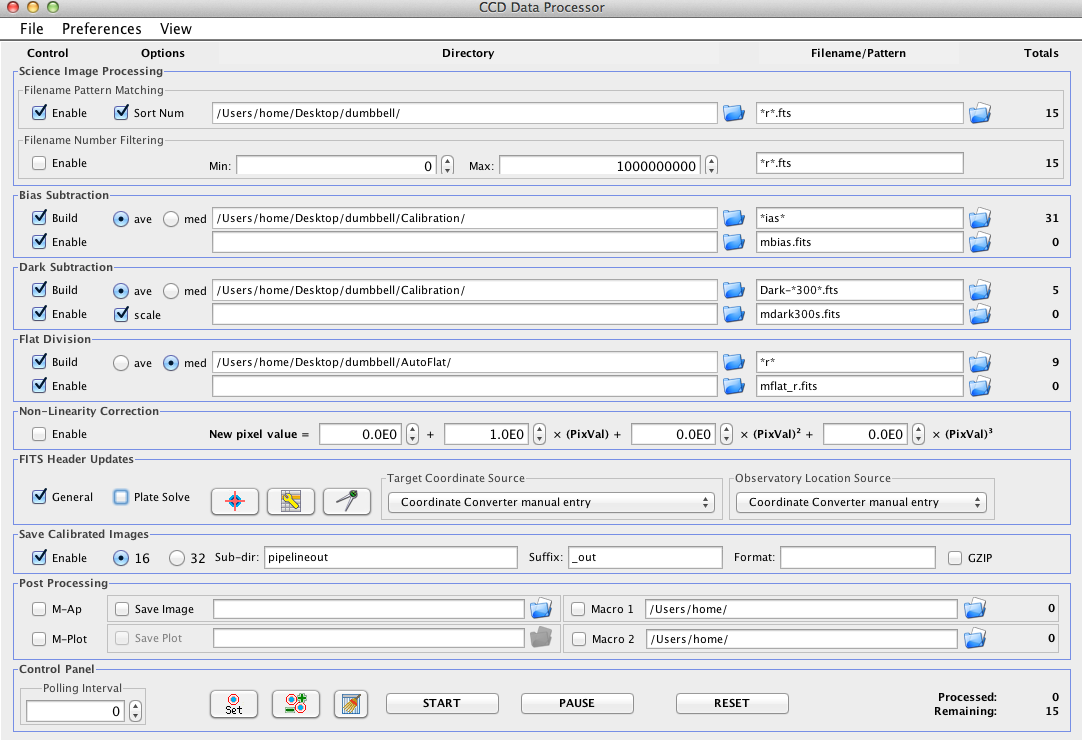
Now we can **reduce** the data. Conceptually, the workflow is: compute “master” bias, dark, and flat frames (one flat for each filter if you have data taken in more than one filter); then correct the data frames, the flat frames, and the master dark by subtracting the bias; then correct the flats and data by subtracting the bias-corrected master dark; then divide the data frames by the corrected flat frames (matching filters), producing the final, reduced images.

In practice, *AstroImageJ* (AIJ) can do this in one single step (albeit with a lot of settings having to be set correctly), at least one single step for one filter.

First, we want to open up the CCD Data Processor window. Click on the “DP” in the floating *AIJ* palette:

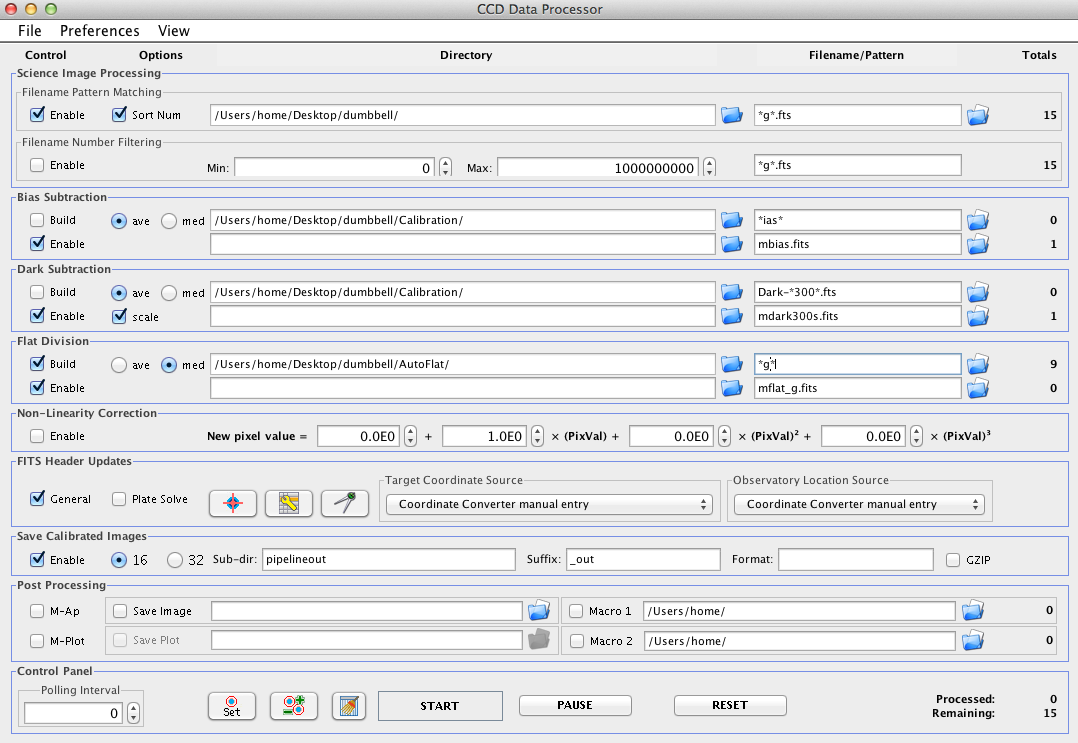


You can close the DP Coordinate Converter window. And in the CCD Data Processor window, indicate the path to the folder with the data files near the top, and use the wildcard character (\*) to identify all data frames taken with a given filter. Fill out the bias, dark, and flat sections, too, pointing in each case to the correct set of files. Check “build” in each case to make the master file, and click “enable” so that master gets used. Here is what the interface looked like for the r’ filter data:



After clicking the “start” button you’ll see messages about the steps that are being run. Then the reduced files (just for the r’ filter) are put in a folder called “pipeline”. Open one up and compare it to the un-reduced version (in the “dumbbell” folder). You should see much less of a diffuse glow and its spatial variation in the reduced image compared to the raw image.

Next, reduce the g’ filter images. You can reuse the master bias and master dark you’ve already made (uncheck the “build” buttons), but you do have to build a new master flat with the g’ filter:



Repeat this process for the H-alpha and H-alpha\_contin filters.

Now you should have sixty new files in the pipeline folder. These are your reduced data.

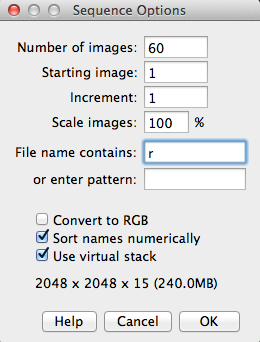
At this point, the integer values in each pixel should ideally contain information only about photons coming into the telescope from the sky. But (a) there will always be some residual noise that isn’t completely calibrated out, (b) dim sources have statistical uncertainties (because you only measure a few photons, and just like an opinion poll that only polls a few people, you may not get a truly representative response), and (c) some of the photons from the sky are noise (light from very distant stars and galaxies that happen to be almost exactly behind the thing you’re trying to observe – astronomical backgrounds – and also light pollution; artificial light scattered off of clouds or haze in the atmosphere – sky backgrounds). So, you should keep these additional sources of error/noise in mind, but after reducing your images, you’ve got a well-calibrated image in which each pixel’s value is a good representation of the relative brightness of the part of the sky imaged onto it.

Image Processing

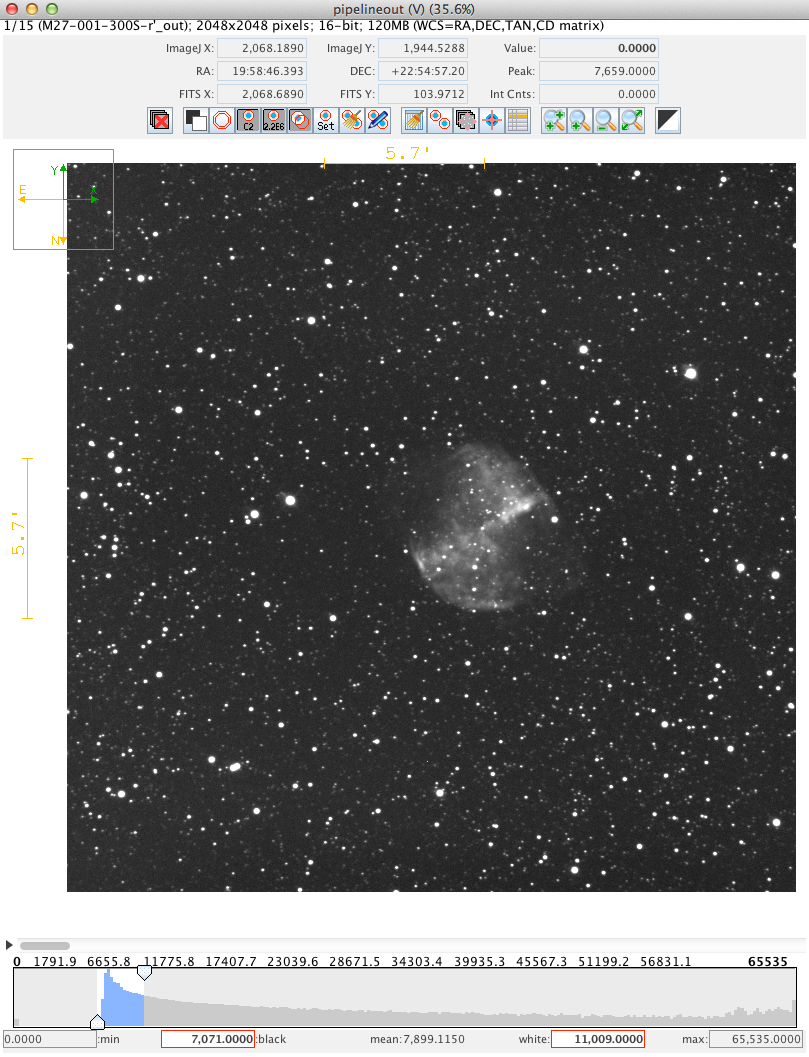
Load in a set of images all taken with the same filter. Here we’ll look at the 15 images of the Dumbbell Nebula taken with the r filter.

Start up *AstroImageJ*, and go to File → Import → Image Sequence.

Navigate to the folder with your reduced science frames. A box will come up, with parameters like “Number of images,” “Starting image,” etc. Go to the box that says “File name contains:” and put in part of the filename that uniquely identifies the files you want (and excludes others). For instance, if you want to open up all the images taken with the r filter, all those images end in “r’\_out.fts” in the folder, so you can put “r” into that box. Make sure to click the “Use virtual stack” clickbox.



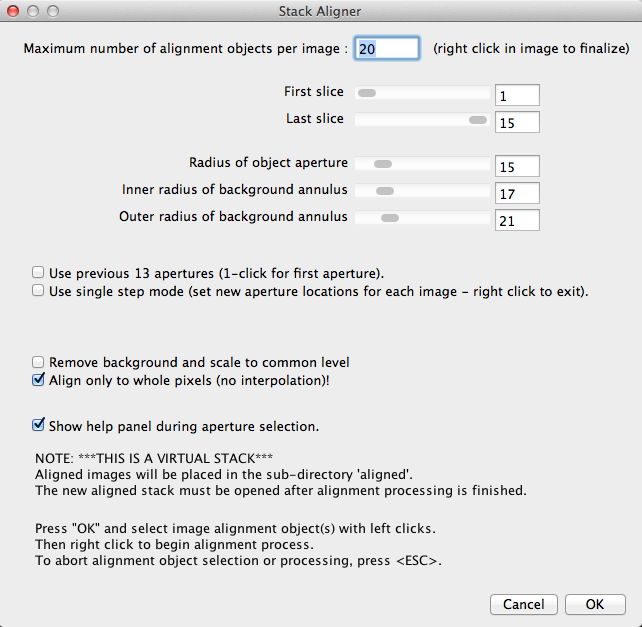
Now all the images should be in a stack. You can flip through the images with the left and right arrows, and there's also a bar towards the bottom of the window that you can drag to change the picture. The name of each picture should be towards the top left of the window, so you can check that the images are really the right ones.



control the grayscale

flip through images

Now, we want to align the images and then add (or average) them together into one master image (in this particular filter). With the stack of images open in *AstroImageJ*, click on the alignment icon above the image (arrow at the top of the image on the previous page points at it). This dialog will open:



Make sure you’ve got the right boxes checked and click “OK” – you’ll then left-click on a selection of stars (your choice) in the image. Pick about a dozen, distributed around the image. Choose stars that are neither very bright nor dim and which aren’t blended with other stars. After you’ve chosen about a dozen, right click anywhere in the image. *AIJ* will align the images (based on the stars you’ve chosen) and create a new folder in which it will put the aligned images.

You can close the stack and now import the newly created images in the folder called “align.” Flip through them and delete any that look bad (e.g. out of focus). The icon with the red X will delete images from the stack.

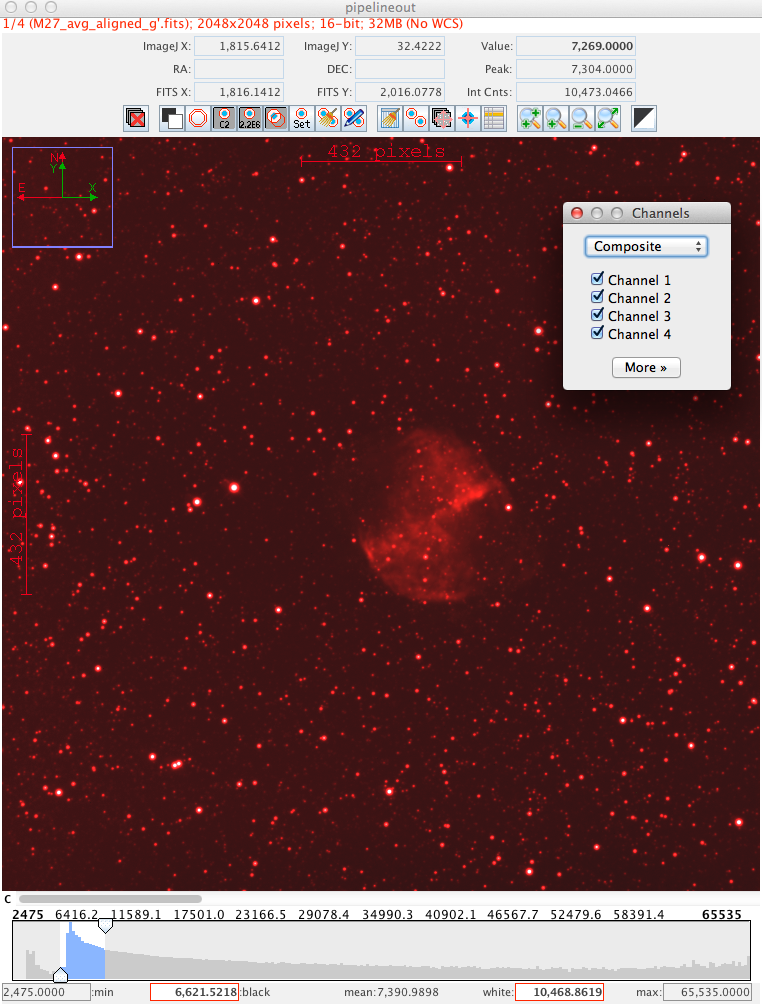
Now we can average the good aligned images in the stack of all the images in a given filter (in the *align* folder). Bring the floating palette of *AIJ* icons to the front, and from the menu at the top of the computer screen choose *images* > *stacks* > *Z project*. You can use the “average intensity” setting from the pull-down menu in the interface that pops up. A single AVG aligned image will be displayed. Save this as a fits file (*file* > *save image slice as FITS*). Add the filter to the file name. Maybe put the word “master” in it too.

Then repeat the process (alignment, read in aligned images, average them, save new master image) for each filter.

At this point, you should have one master image for each filter.

Open these master images in a stack (same way you’ve done it before: *file* > *import* > *image sequence*). For the M27 data, this would be four images: r, g, Halpha, and Halpha\_contin.

In the window containing your stack, go to the menu at the top of the screen *Color* *→ Make Composite color image*. A box will come up, and check “Display Mode: Composite.” Then another box will come up, with the title “Channels.” It should have a drop-down box, four channels with check-boxes, and a button that says “More.”



Step through the images one at a time, and uncheck the boxes so that each image in the stack is assigned to a unique channel. Then for each channel, you can click on the “more” button and choose a color for it. At this point you can also use the grayscale sliders to adjust the brightness of that channel.

Note that the g filter is green, r is red, and – unfortunately – Halpha and Halpha\_contin are narrow filters that are also in the red. So, these data aren’t really well distributed through the spectrum. We have no real blue information. So, it’ll be hard to make a color image that’s got realistic colors in this case.

Now you have your color-coded, brightness adjusted master images. You can combine them by checking the appropriate boxes. To save them as a single image use the “more” button and the “convert to RGB” option. Then you can save the resulting, combined image in any number of formats using the file menu.

You can go through the process again with these observations of the Whirlpool Galaxy, M51:

<http://astro.swarthmore.edu/~cohen/telescope/M51.zip>