Tutorial VI
Astronomy Image Analysis-I:
DS9 and its Pythonic Approach

By Lauren Higgins

Task I

4 steps

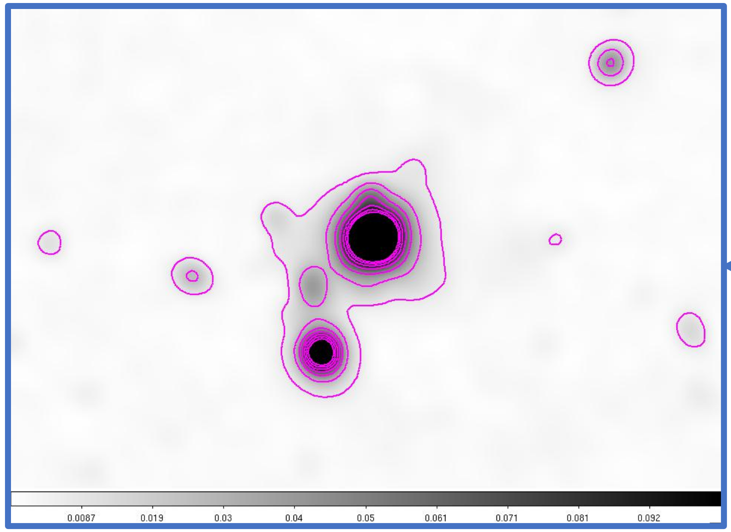
Activities I & II

Task I: Activities I and II

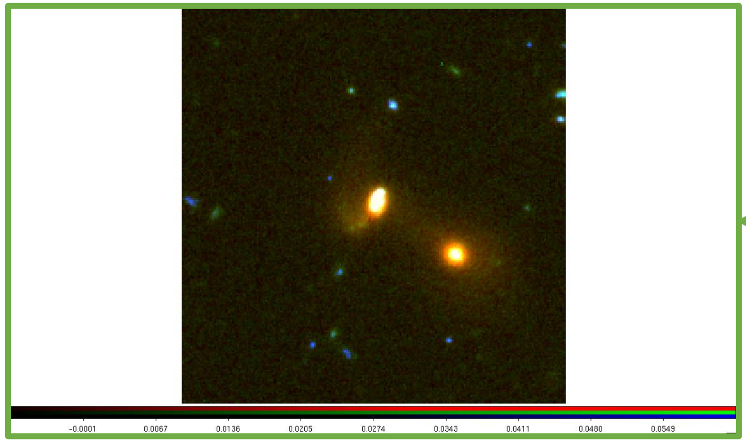
Activity I.
The MIN-MAX interval is can be a quick way to make an image when in a time crunch. It could be too simple and not accurately account for any counting error in the measurement of photons. The percent scaling option gives the user control over how much data within each pixel contributes to the final image and can vary based on the type of image a user wants.

Activity II.
A BoxCar filter smooths an image by equally weighing a rectangular area of neighboring pixels[4]. A “TopHat” filter extracts small features from an image[3]. A ”Gaussian” filter fits a gaussian profile to each pixel in an image[1, 2, 4]. Each filtering type is dependent on the size of the kernel used for the filter[1].

Task I: Two Images



1.3.8.
I created the image to the left with personalized zoom, scaling, contours, colors, and smoothing. I doubled the zoom view, used Linear 99.5% scaling, changed the contours have a level of 8 and smoothness of 14, inverted the black and white color, and I used gaussian smoothing with a radius of 14 and sigma of 7.



Step 4.
I used [5] to tell me which files corresponded to red, blue, and green. This is zoomed out for the entire image to fit the ds9 window.

Task II

Section A (*large_mosaic & postage_stamp*): 6 Steps

Section B (*image_cube*) : 3 Steps

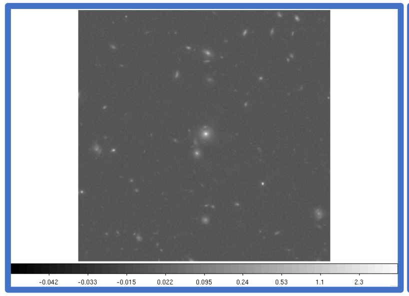
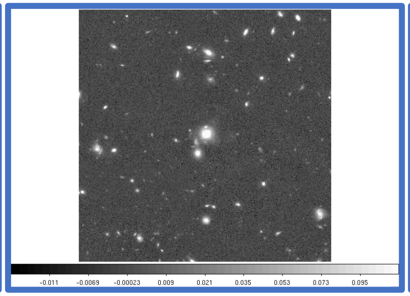
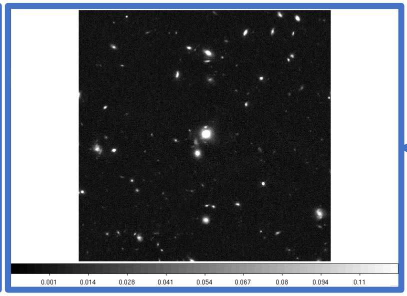
Section C (*for_RGB*) : 2 Steps

Task II:
Section A – Step 1

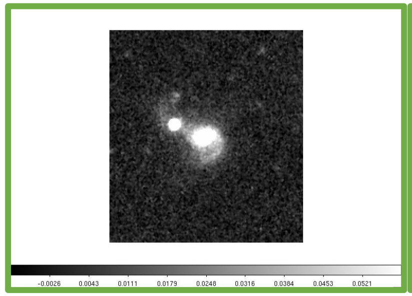
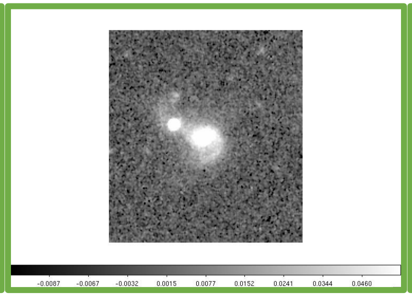
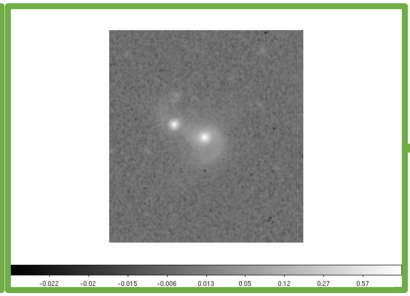
Step 1.
The large_mosaic.fits' pixel scale in arcseconds per pixel is ~6e-2"/pix. Since the image has 1000 rows of pixels and 1000 columns of pixels, then the image span is approximately 60" by 60", giving a total area of ~3600".

The postage_stamp.fits' pixel scale in arcseconds per pixel is also ~6e-2"/pix. Since the image has 183 rows of pixels and 201 columns of pixels, then the image span is approximately 11" by 12", giving a total area of ~132".

Task II:
Section A – Step 2

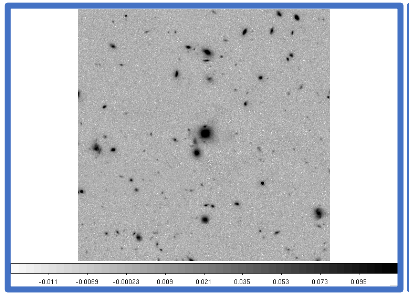
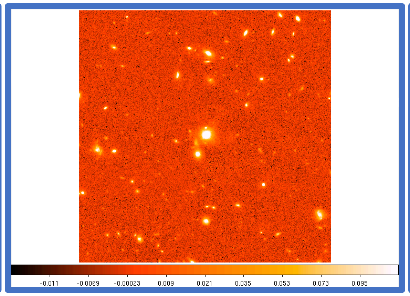
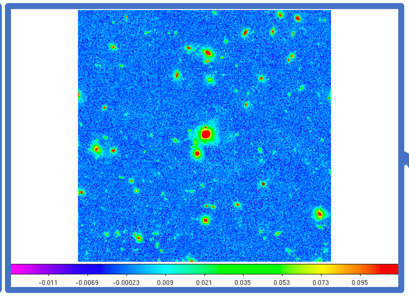




Step 2. large_mosaic.fits
Here are three trials for the large_mosaic.fits image. The first is MINMAX log scale. The second is 99.5% square root. The third is 99.5% linear scale. I believe the second image to be the best.

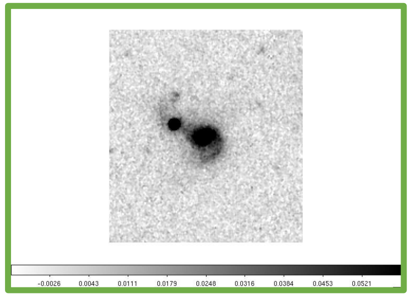
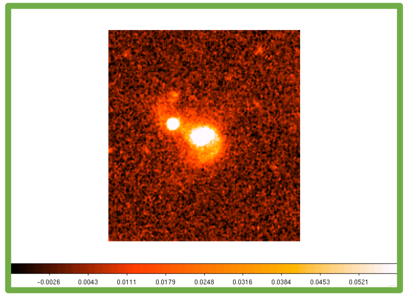
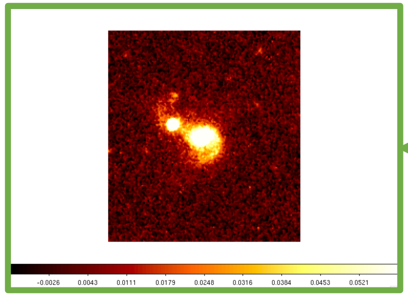




Step 2. postage_stamp.fits
Here are three trials for the postage_stamp image. The first is 98% linear scale. The second is 98% square root. The third is MINMAX log scale. I believe the first image to be the best.

Task II:
Section A – Step 3

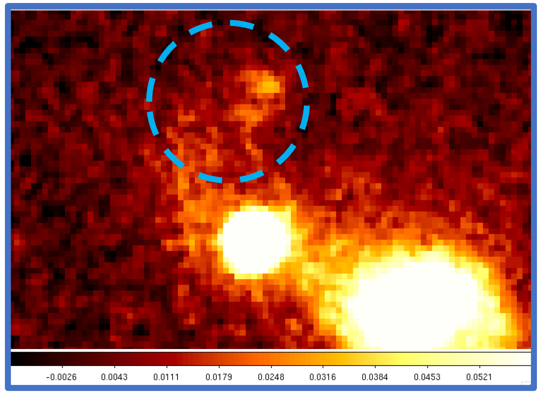




Step 3. large_mosaic.fits
Here are three trials for the large_mosaic.fits image. The first is an inverted grey color map. The second is a heat color map. The third is a rainbow color map. I believe the first image is the best I have made thus far. I like the inverted grey color map because my eyes go immediately to where galaxies live in the image. Furthermore, there is structure shown in each galaxy without exaggerating any data.

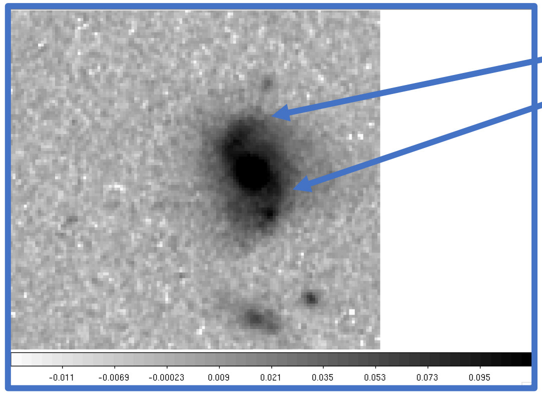




Step 3. postage_stamp.fits
Here are three trials for the postage_stamp.fits . The first is an inverted grey color map. The second is a heat color map. The third is a bb color map. I believe the third image is the best I have made thus far. The bright spots really stand out from the noise and structure is visible.

Task II:
Section A – Step 4



Zoom on post postage_stamp.fits

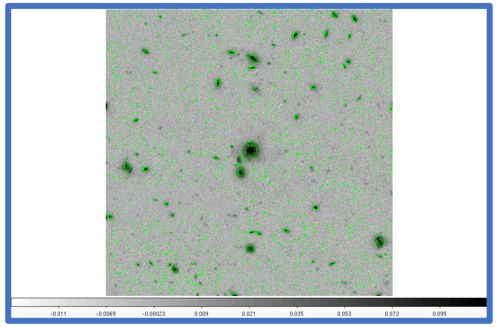


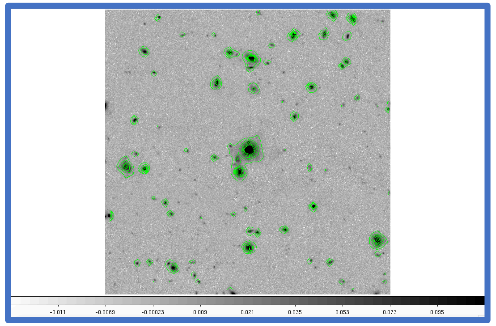
Zoom on post large_mosaic.fits

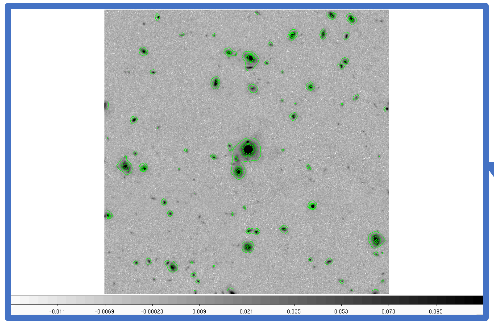
Step 4.
The top image is a zoom on what appears to be an interacting/merging galaxy system. I can see a tidal tail curling up and around the smaller bulge, shown in the dashed circle.

The bottom image appears to be a spiral bar galaxy. There are what looks like two spiral arms. There is also a darker area in the center of the galaxy, indicating a bright bulge.

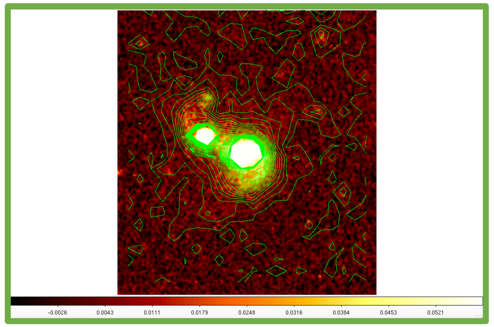
Task II:
Section A – Step 5

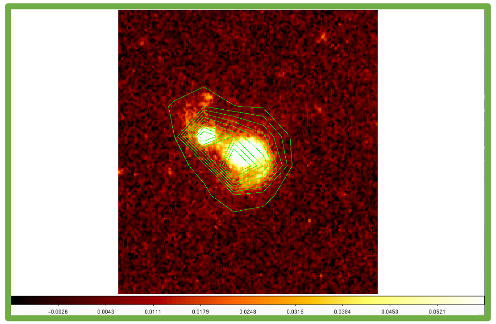


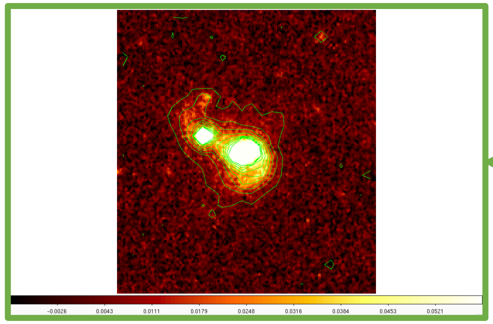




Step 5. large_mosaic.fits
Here are three trials for the large_mosaic.fits image. The first has a very low smoothing value and a low amount of levels. The second still has low level value but has high smoothing. The third, and best, has 6 levels and a smoothing value of 11.

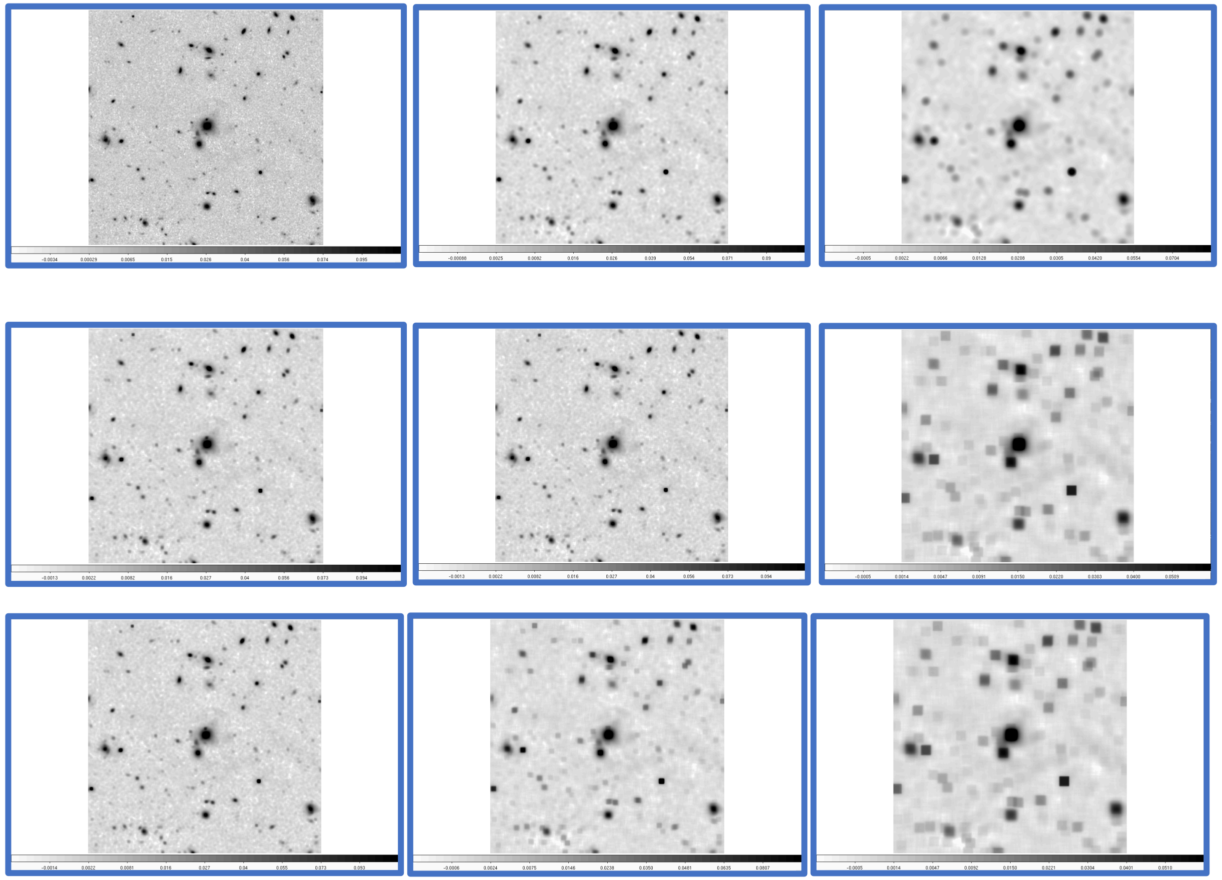






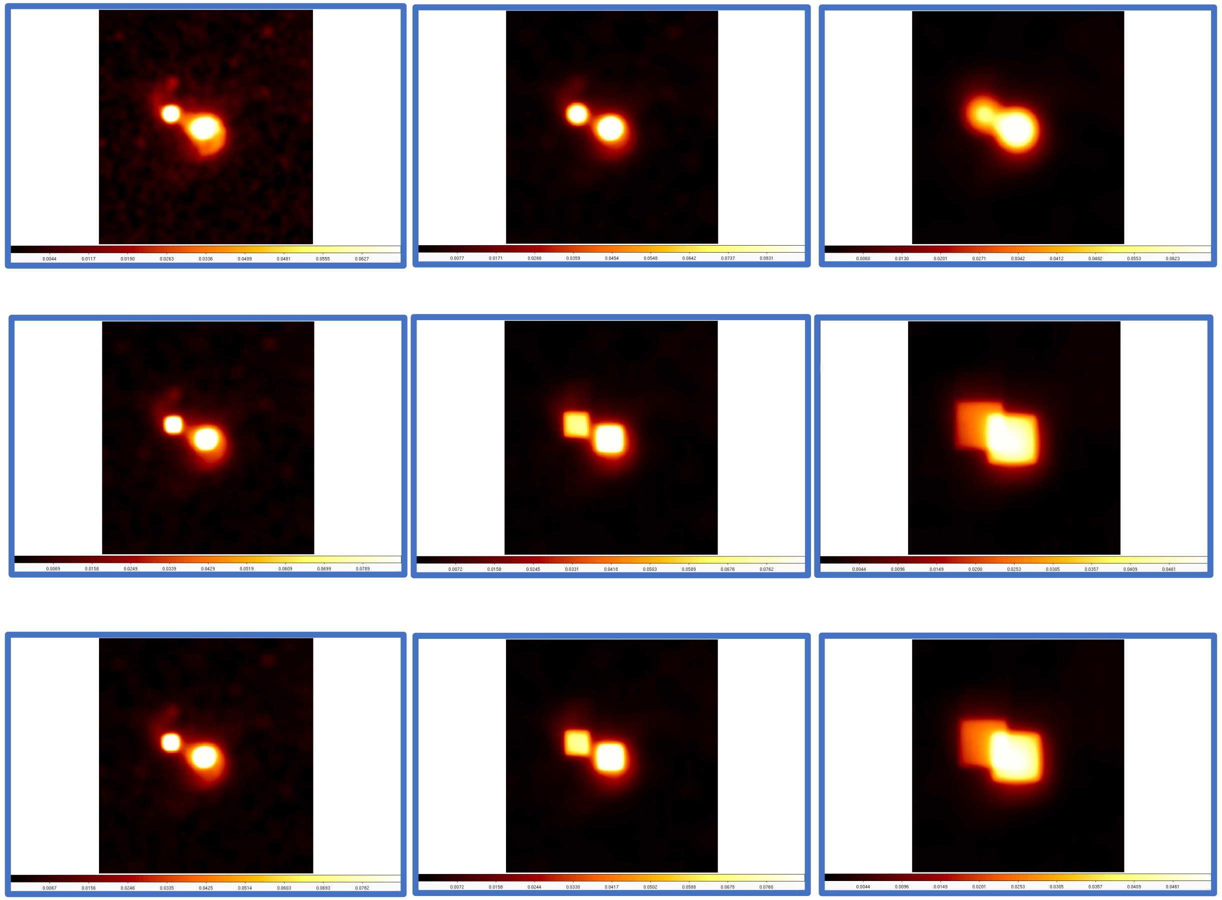
Step 5. postage_stamp.fits
Here are three trials for the postage_stamp.fits . The first has a very high level value and very low smoothing. The second has a low level value and very high smoothing. The third, and best, has 11 levels and a smoothing value of 7.

Task II:
Section A – Step 6 (*large_mosaic*)



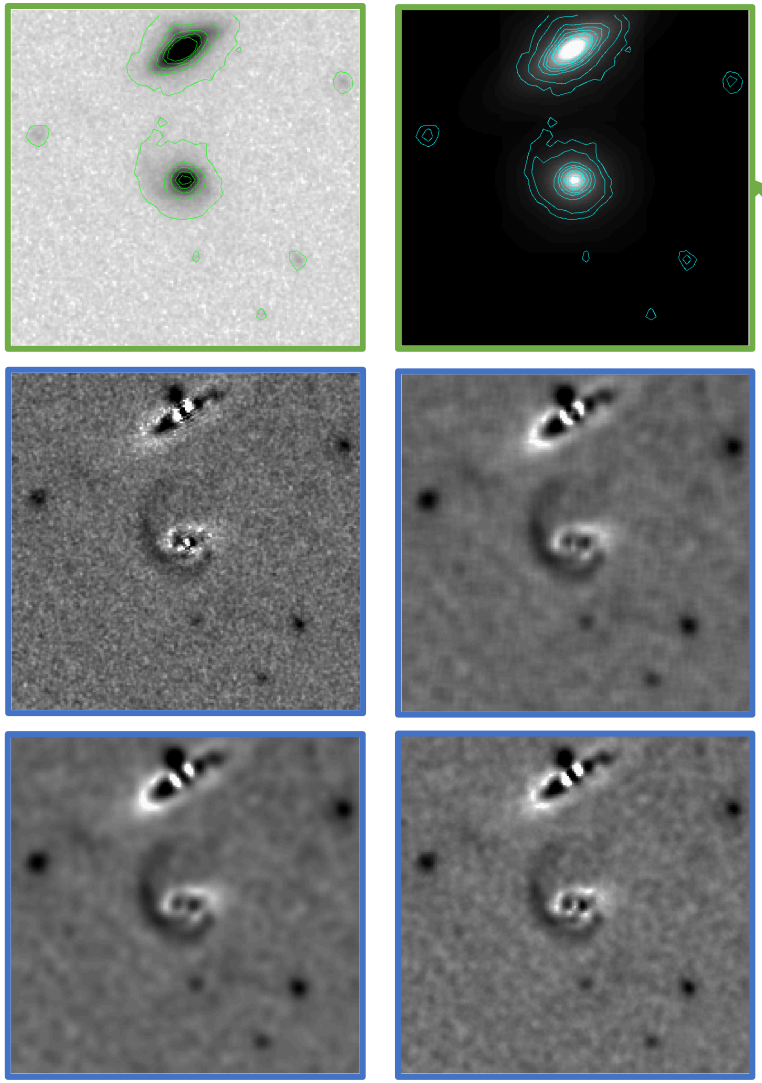
Step 6.
The first row has three Gaussian filter examples with radii of 5, 10, and 20. As I increase the radius value the small features of the image disappear and the bulges of each galaxy get brighter and more rounded. The second row has three BoxCar filter examples with radii of 5, 10, and 20. The third row contains three TopHat filter examples with radii of 5, 10, and 20. Both the BoxCar and TopHat filters become more rectangular as I increase the smoothing. This makes sense because the two functions that define their shape are square. TopHat smoothes out slightly more pixels than BoxCar.

Task II:
Section A – Step 6 *(postage_stamp)*



Step 6.
The first row has three Gaussian filter examples with radii of 5, 10, and 20. As I increase the radius value the small features of the image disappear and the bulges of each galaxy get brighter and more rounded. The second row has three BoxCar filter examples with radii of 5, 10, and 20. The third row contains three TopHat filter examples with radii of 5, 10, and 20. Both the BoxCar and TopHat filters become more rectangular as I increase the smoothing. This makes sense because the two functions that define their shape are square. TopHat smoothes out slightly more pixels than BoxCar.

Task II:
Section B – Steps 1 - 3

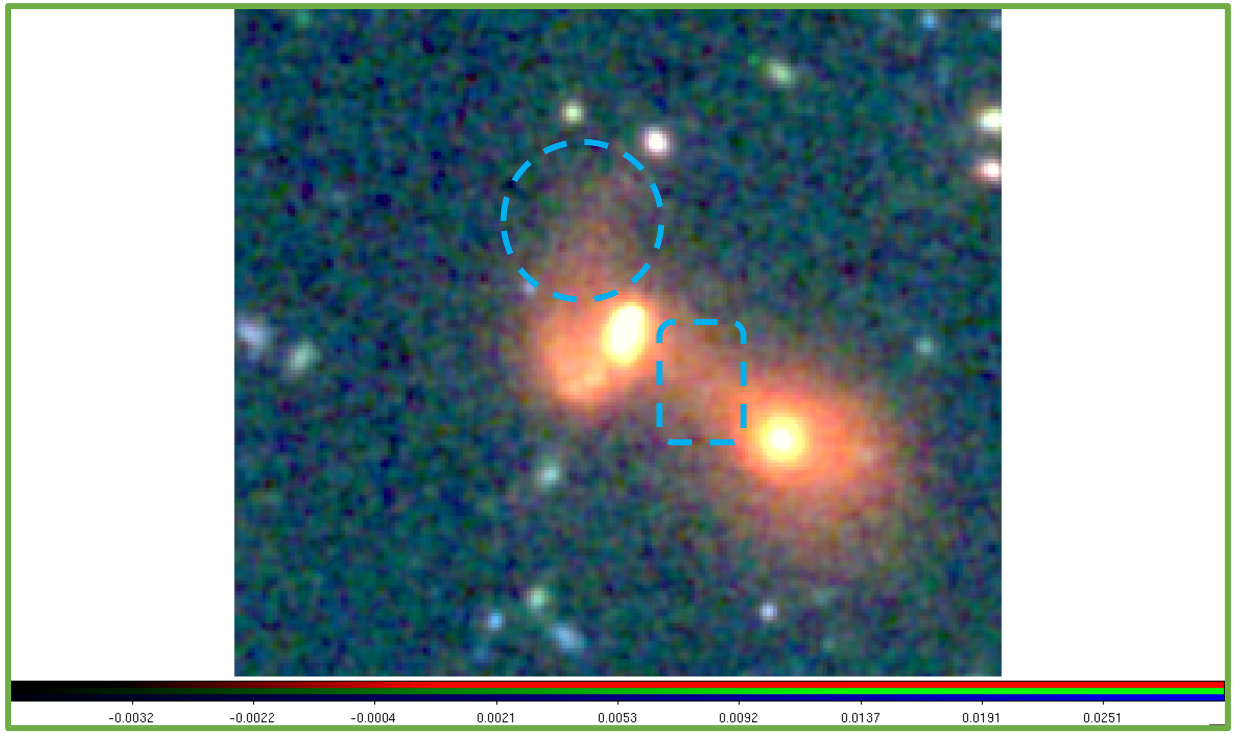


Step 1.
I opened the image_cube.fits using the instructions from 1.3.1 of the tutorial. I used 99.5%, sqrt, and grey color map for each tile and I inverted the colors for extensions one and three. Extension two did not translate well to an inverted colormap.

Step 2.
I created contours for the first image and then copied and pasted them onto the second image.

Step 3.
I smoothed image number 3. The upper left of the images outlined in blue is the unsmoothed image. The upper right of those images is the BoxCar filter of size 3. I notice that the noise in the image is washed out and there seems to be a double nucleus in the center of the central galaxy and perhaps a multi system interaction at the top of the image.
The bottom left image is a Gaussian with a radius of 5 and the bottom right image is a TopHat with a radius of 2. There seems to be an optimal radius value for an image that best illultrates the two interesting features.

Task II:
Section C – Steps 1 & 2



Step 1.
The “F” in each filter name stands for filter and the “W” in each filter name stands for wide. The filter f160w corresponds to the longest of the wavelengths, so in relation to the other two it is red, filter f814w is the mid wavelength filter, so it corresponds to green and filter f606w is the shortest wavelength filter, so it corresponds to blue.

Step 2.
Through creating this image I realized that it is is an RBG image of the postage_stamp.fits image. There appears to be two interacting galaxies and a tidal tail within the dashed circle. It is also more apparent in this image that there is some matter shared between the two galaxies (shown in the dashed rectangle). There are also many other small galaxies in the same area. These could be physically smaller than the central galaxies or just farther away.

Task III

Activity III

4 steps

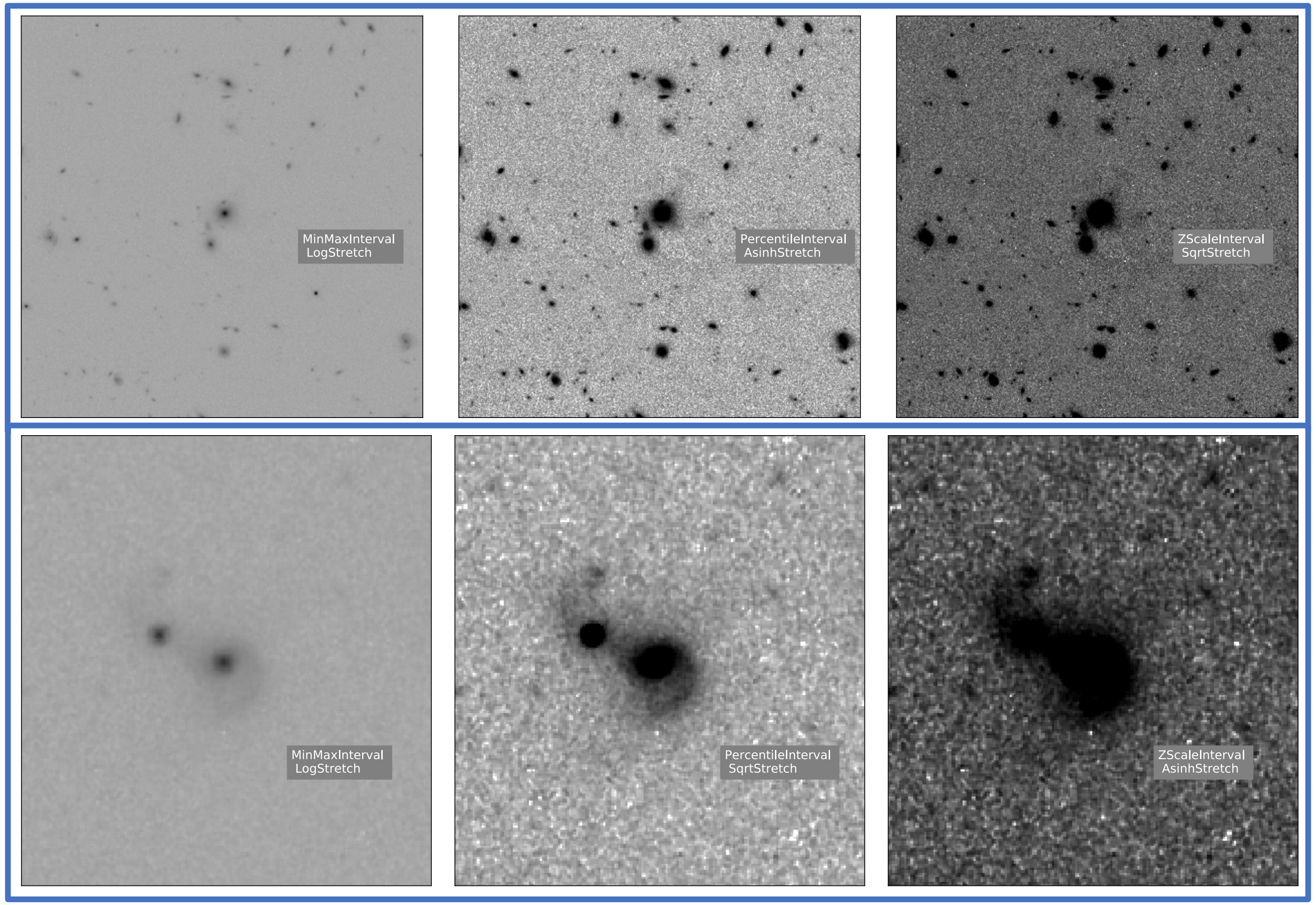
Task III: Activity III

```
In [12]: print(image_header['PC2_2'],'\n' , image_header['NAXIS2'],'\n' , wcs_image)
1.666667e-05
 1000
 WCS Transformation

This transformation has 2 pixel and 2 world dimensions

Array shape (Numpy order): (1000, 1000)

Pixel Dim  Data size  Bounds
        0       1000  None
        1       1000  None

World Dim  Physical Type  Units
        0  pos.eq.ra      deg
        1  pos.eq.dec     deg

Correlation between pixel and world axes:

           Pixel Dim
World Dim    0    1
        0  yes  yes
        1  yes  yes
```

Activity III. I printed the following things:

- PC2_2 value “1.666667e-05”
- NAXIS2 value “1000”
- wcs of the image

Task III: Step 1

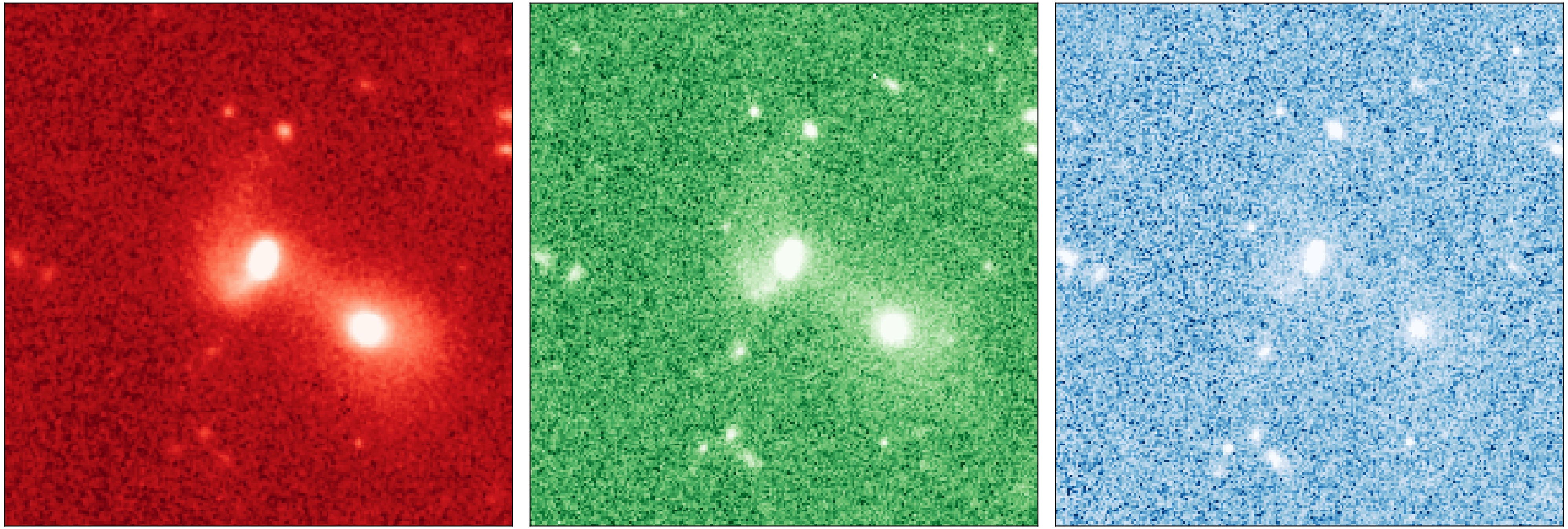


The top image is of the large_mosaic.fits and the bottom image postage_stamp.fits. The process I used for these images is outlined in my code file Tutorial_VI_TaskIII_1.py[6].

Task III: Step 2

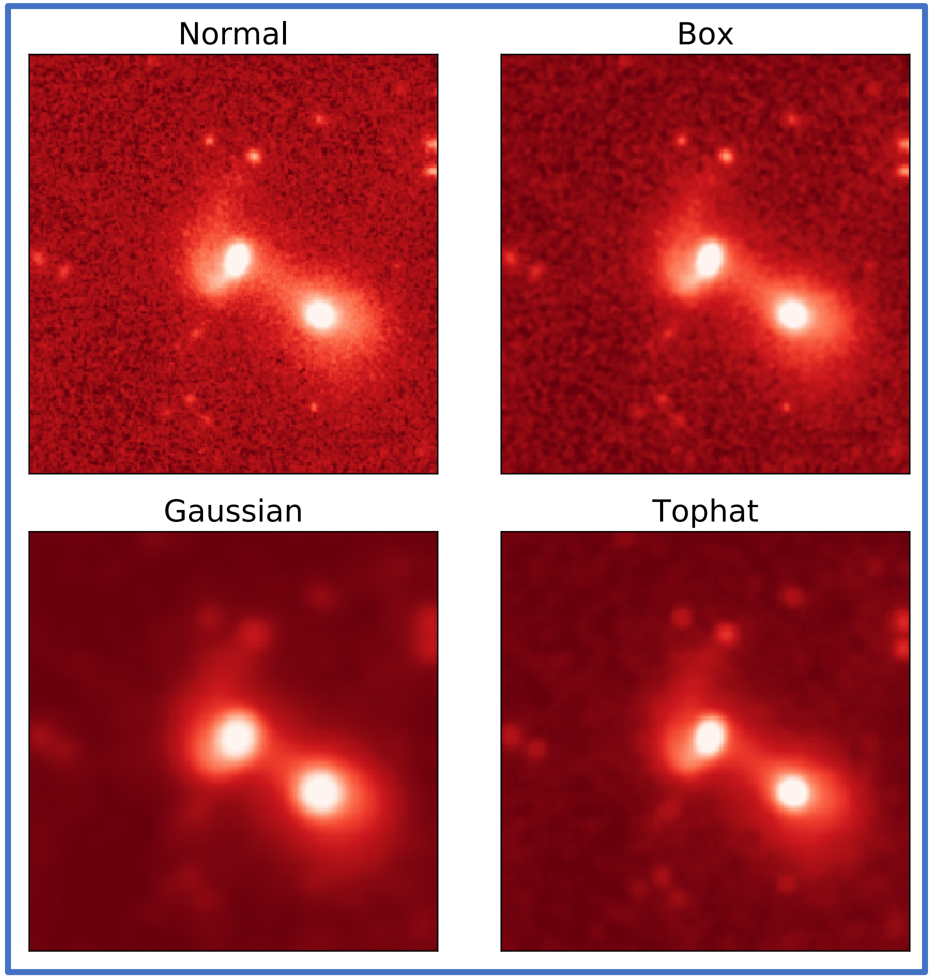
This is the image_cube.fits file. Please see the file Tutorial_VI_TaskIII_2.py for how I created this image.

Task III: Step 3



This is the image_cube.fits file. Please see the file Tutorial_VI_TaskIII_3.py for how I created this image.

Task III: Step 4



This is the "*_f160w.fits" file with different convolving filters as denoted by the title of the panel. Please see the file Tutorial_VI_TaskIII_4.py for the code I used to make this image[7].

Collaborators

- Task I: Grace
- Task II: none
- Task III: Greg

Sources

[1] http://ds9.si.edu/doc/ref/how.html

[2] https://en.wikipedia.org/wiki/Gaussian_filter

[3]http://www.theobjects.com/dragonfly/dfhelp//Content/05_Image%20Processing/Morphology%20Filters.htm

[4] https://courses.cs.washington.edu/courses/cse576/book/ch5.pdf

[5] https://hubblesite.org/contents/media/images/2012/17/3011-Image.html?news=true

[6] https://stackoverflow.com/questions/17086847/box-around-text-in-matplotlib

[7] https://matplotlib.org/devdocs/gallery/subplots_axes_and_figures/subplots_demo.html