- **Activities**

## In Practice: Basic CCD Image Treatment

Here we seek to apply this basic CCD imaging treatment sequence to real data. You will perform basic dark subtraction and flat fielding of target images, and then shift these images into alignment to produce an effectively "deeper" integration on the source. The science target is the nearby starburst galaxy M 82. You are provided with dark images (dark1.fits – dark7.fits), flat field images (flati1.fits, flati2.fits), and images of the galaxy (m82i1.fits – m82i3.fits). The target and flat field images are in the "I" filter.

1. Identify the exposure time of each image using the HSELECT task in *IRAF*, and record these values in the table below.

```
cl> hselect dark*fits $I,EXPTIME yes
cl> hselect flat*fits $I,EXPTIME yes
cl> hselect m82*fits $I,EXPTIME yes
```

| – Image Characteristics – | |
|---|---|
| **Image** | **Exposure Time** |
| dark1.fits | |
| dark2.fits | |
| dark3.fits | |
| dark4.fits | |
| dark5.fits | |
| dark6.fits | |
| dark7.fits | |
| flati1.fits | |
| flati2.fits | |
| m82i1.fits | |
| m82i2.fits | |
| m82i3.fits | |

2. Recalling the rules for integration times for dark, flat and science images, identify which dark images to use for both:

   (a) Dark image(s) to use for flat field images: ________________________

   (b) Dark image(s) to use for science images: ________________________

3. Average together dark images with the same integration time using the *IRAF* task IMCOMBINE (replace "name" below with the appropriate dark images to be combined):

```
cl> imcombine input=name1,name2,etc. output=dark reject=crreject hsigma=3
```

Now examine the resultant image(s) using DS9 to assure that everything went as expected.

4. Subtract the relevant dark image from each of the flat field and target images using the *IRAF* task IMARITH.

5. Next we must combine our (dark-subtracted) flat field images; use the *IRAF* task IMCOMBINE with no outlier rejection:

   ```
   cl> imcombine input=name1,name2 output=flat reject=none
   ```

   Examine the resultant image using DS9 to assure that everything went as expected.

6. Normalize this flat field image by dividing by the mean of the central regions; in your DS9 display it will be obvious that the edges of the CCD image contain undesirable artifacts. Identify a range of usable X- and Y-coordinates that avoids these edge effects and find the mean using IMSTAT. Then use IMARITH to divide the flat field image by this number.
   What mean value did you find for your image? ________________________
   Using IMSTAT on the central regions, check that the MEAN value of this new image is in fact very very close to 1.

7. Apply the flat-field correction to the (dark-subtracted) science images by dividing them by the final flat field image; use the *IRAF* task IMARITH.

8. Subtract the mean sky background from each of the science images using the *IRAF* task IMARITH; determine this value for each image by using IMSTAT on various regions that do not contain any evidence of signal from the galaxy.

9. There is one foreground star in this sequence of images; using it, estimate a rough shift for any two of the images that will bring the 3 science images close to alignment. For reference, I have determined the coordinates of this star to be approximately as follows:

   ```
   m82i1.fits: (x,y) = (609.109, 399.370)
   m82i2.fits: (x,y) = (608.204, 393.119)
   m82i3.fits: (x,y) = (612.550, 388.347)
   ```

   Use the *IRAF* task IMSHIFT for this purpose.

10. Co-add these (shifted) images to produce a single (and effectively deeper) image of the target using the *IRAF* task IMCOMBINE with outlier rejection:

```
cl> imcombine input=name1,name2,etc. output=m82_final reject=crreject hsigma=3
```

Examine the resultant image using DS9 to assure that everything went as expected.

11. Finally, trim the edge effects off of the final image to produce an aesthetically pleasing final product; identify the region to be trimmed using DS9 and use the IRAF task IMCOPY (replace "name" with the appropriate image and "x" and "y" with appropriate pixel dimensions):

```
cl> imcopy input=name[x:x,y:y] output=final
```